

**UNIVERSIDAD TECNOLÓGICA**  
**ISRAEL**

**FACULTAD DE SISTEMAS INFORMÁTICOS**

**Sistema de Ingresos y Egresos para determinar el  
Impuesto a la Renta en la web para personas naturales  
no obligadas a llevar contabilidad**

**Estudiante**

**Alfonso Rafael Chumbi Mendieta**

**Tutor**

**Ing. Leopoldo Pauta**

**Cuenca Ecuador**

**Noviembre 2011**

**UNIVERSIDAD TECNOLÓGICA**

**ISRAEL**

**FACULTAD DE SISTEMAS INFORMÁTICOS**

**CERTIFICADO DE RESPONSABILIDAD**

Yo Ing. Leopoldo Pauta, certifico que el Señor Alfonso Rafael Chumbi Mendieta con C.C. No. 0101885309 realizó la presente tesis con título “SISTEMA DE INGRESOS Y EGRESOS PARA DETERMINAR EL IMPUESTO A LA RENTA EN LA WEB PARA PERSONAS NATURALES NO OBLIGADAS A LLEVAR CONTABILIDAD”, y que es autor intelectual del mismo, que es original, autentica y personal.

---

**Ing. Leopoldo Pauta**

**UNIVERSIDAD TECNOLÓGICA**  
**ISRAEL**

# FACULTAD DE SISTEMAS INFORMÁTICOS

## CERTIFICADO DE AUTORÍA

El documento de tesis con título “**SISTEMA DE INGRESOS Y EGRESOS PARA DETERMINAR EL IMPUESTO A LA RENTA EN LA WEB PARA PERSONAS NATURALES NO OBLIGADAS A LLEVAR CONTABILIDAD**” ha sido desarrollado por Alfonso Rafael Chumbi Mendieta con C.C. No. 0101885309 persona que posee los derechos de autoría y responsabilidad, restringiéndose la copia o utilización de cada uno de los productos de esta tesis sin previa autorización.

---

**Alfonso Rafael Chumbi Mendieta**

## **DEDICATORIA**

El presente trabajo se lo dedico a mi esposa, quien me brindó su amor, cariño, estímulo, confianza, constante apoyo, comprensión y paciencia para que pudiera terminar con éxito una etapa más de mi vida.

No tengo palabras para expresarte mi agradecimiento.

Te amo querida esposa.

A mi adorado hijo Alexander y a mi pequeña Rafaela, quienes me prestaron el tiempo que les pertenecía y me

motivaron siempre con su alegría para culminar mis estudios.

### **AGRADECIMIENTO**

Un profundo agradecimiento a la Universidad Israel, la Facultad, sus autoridades y todos los profesores de Sistemas Informáticos.

De manera muy especial agradezco al Ing. Leopoldo Pauta,  
Director de tesis, quien con su disposición me ha orientado con sus conocimientos para el desarrollo y la culminación exitosa de la presente tesis.

### **RESUMEN**

El presente estudio se concentra en el cálculo del Impuesto a la Renta de Personas Naturales no obligadas a llevar contabilidad, en vista de que, este sector esta desatendido por los programas informáticos actuales.

Por la complejidad del cálculo el Impuesto a la Renta de las personas naturales, es necesario de la presencia de un asesor contable y/o tributario para efectuarlo, labor que sería reemplazada por la herramienta informática propuesta en este proyecto, con el consecuente ahorro de tiempo y dinero, minimizando el riesgo de errores u omisiones dados por el desconocimiento de la aplicación de la Ley y su Reglamento, o por errores involuntarios en el cálculo.

Se utilizó investigación de campo, aplicada y bibliográfica, recurriendo a entrevistas, cuestionarios, encuestas y observación para la recopilación de información; además de la aplicación de teorías y de disposiciones legales, con la finalidad de resolver el problema propuesto en este proyecto.

En cuanto a la metodología informática se utiliza la “Metodología **Orientada a Objetos**”, ya que nos permite aplicar varias técnicas, como herencia, modularidad, polimorfismo, y



encapsulamiento. Utilizando la plataforma de programación Visual.net 2005, y como Base de datos SQL Server 2005.

Se empleó el Proceso Unificado puesto que es un marco de desarrollo interactivo e incremental compuesto de cuatro fases denominadas: Inicio, Elaboración, Construcción y Transición. Cada una de estas fases estuvo dividida en una serie de iteraciones dando como resultado un producto de excelente calidad.

### **SUMMARY**

The present study concentrates on the tax computation to the Revenue of the natural persons not forced on taking accounting, considering that, this sector is neglected by the computer current programs.

For the complexity of the calculation the Tax to the Revenue of the natural persons, is necessary of the presence of a countable and/or tributary adviser to carry it out, work that would be replaced by the computer tool proposed in this project, with the consistent saving of time and money, minimizing the risk of errors or omissions given by the ignorance of the application of the Law and his regulation, or by involuntary errors in the calculation.

Bibliographical field investigation is used, applied and, resorting to interviews, questionnaires, polls and observation for the compilation of information in addition to the application of theories and laws, with the purpose of solving the problem proposed in this project.

As for the computer methodology there is used the “ Methodology guided to Objects ”, since it allows us to use several skills, like heredity, modulate, polymorphism, and

encapsulate. Using the platform of programming Visual.net 2005, and as Database SQL Server 2005.

The Process that followed is the Unified Process since it is a frame of development interactive and incremental composed of four named phases: Beginning, Making, Construction and Transition. Each of these phases was divided in a series of iterations giving like turned out a product of excellent quality.

## **LISTA DE ANEXOS**

ANEXO 1 Bases Del Anteproyecto

ANEXO 2 Guía Práctica para la Declaración del Impuesto a la Renta para Personas Naturales No Obligadas a Llevar Contabilidad



## LISTA DE CUADROS

	PÁGINA
CUADRO 1: Presupuesto General del Estado, Porcentaje de Fuentes de Ingresos	3
CUADRO 2: Recaudación de Impuestos de Personas Naturales 2008	6
CUADRO 3: Recaudación del Servicio de Rentas Internas Frente al Presupuesto General del Estado 2008	7
CUADRO 4: Resumen de problemas	10
GRÁFICO 1: Nivel de Conocimiento sobre Impuesto a la Renta	11
GRÁFICO 2: Recaudación enero-marzo 2010	12
GRÁFICO 3: Crecimiento nominal de recaudación enero-marzo 2010	13
CUADRO 5: CUADRO para el pago de multas	14
CUADRO 6: Tasa de Interés Trimestrales por Mora Tributaria	15
CUADRO 7: Fechas de pago	16
CUADRO8: Matriz de prioridades: Plataforma de desarrollo	26
CUADRO 9: Matriz de Ponderación: Plataforma de desarrollo	26
CUADRO10: Matriz de prioridades: Base de datos	27
CUADRO 11: Matriz de Ponderación: Base de datos	28
CUADRO 12: Matriz de prioridades: Ciclo de vida	29
CUADRO13: Matriz de Ponderación: Ciclo de vida	29
CUADRO14: Hardware del servidor	30
CUADRO 15: Matriz de prioridades: Sistema operativo	31
CUADRO 16: Matriz de Ponderación: Sistema operativo	31
CUADRO 17: Relación costo beneficio	33
GRÁFICO 4: Fases del RUP	56

CUADRO 18: Ciclo de vida del proyecto	70
CUADRO 19: Roles y responsabilidades	84
CUADRO 20: Costo personal	85
CUADRO 21: Descripción de los equipos requeridos	85
CUADRO 22 Descripción de viajes	86
CUADRO 23: Descripción de Insumos y Materiales	86
CUADRO 24:Descripción Servicios Técnicos	86
CUADRO 25: Costo total del proyecto	87
CUADRO 26: Plan de fases	88
CUADRO 27: Hitos de las fases	88
CUADRO 28: Calendario del Proyecto	90
CUADRO 29: Sentencia que define el problema	117
CUADRO 30: Sentencia que define la posición del Producto	117
CUADRO 31: Resumen de Stakeholders	118
CUADRO 32: Resumen de Usuarios	119
CUADRO 33: Perfil de Stakeholders	120
CUADRO 34: Perfiles de Usuario	121
CUADRO 35: Resumen de características	122
CUADRO 36: Requisitos del sistema	123
CUADRO 37: Requisitos de Desempeño	124
CUADRO 38: Requisitos de Entorno	124
CUADRO 39: Especificaciones del caso de uso: Registrar ingresos	125
CUADRO 40: Especificaciones del caso de uso: Registrar gastos	125
CUADRO 41: Especificaciones del caso de uso: Calcular Impuesto a la Renta	126

# 1. INTRODUCCIÓN

## 1.1. ANTECEDENTES

En la República del Ecuador se realiza una distinción importante en el tratamiento del Impuesto a la Renta, tanto para personas naturales como para las sociedades.

En el caso de las personas naturales obligadas a llevar contabilidad y las sociedades, el cálculo del Impuesto a la Renta viene dado por los resultados que arroja la contabilidad.

En tanto que la persona natural no obligada a llevar contabilidad se limita a realizar el cálculo de forma manual, y a veces sin el respectivo respaldo de un contador, aspecto por el cual se sujeta a la posibilidad de incurrir en errores, los mismos que serían controlados mediante el uso de un software creado para el efecto.

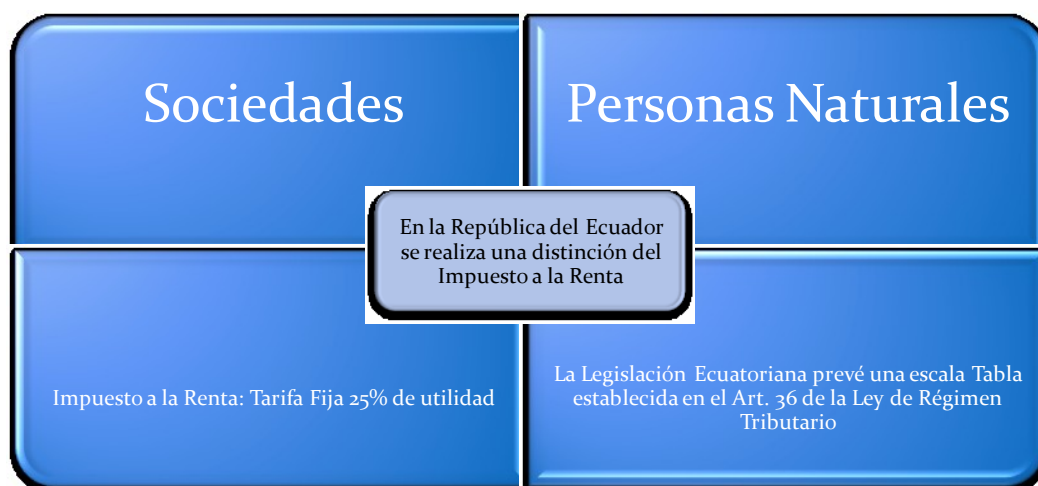
La Legislación Ecuatoriana provee una escala ascendente para la determinación del Impuesto a la Renta de personas naturales, realizando cálculos específicos por cada base imponible de la tabla establecida en el Art. 36 de la Ley de Régimen Tributario, la misma que se actualiza anualmente.

A demás el cálculo del Impuesto se basa en un cúmulo de variables que complican el mismo, razón por la cual sería de suma utilidad un programa informático que contenga dichas variables y simplifique los cálculos.

El SRI tiene un modelo de formulario que se llena y se envía por Internet, sin embargo éste

es muy escueto y sin las explicaciones del caso, que permitan a una persona sin conocimientos contables o tributarios realizar su declaración con la seguridad de cumplir con las exigencias legales y reglamentarias previstas. Se limita únicamente a solicitar datos que son resultados de procedimientos anteriores, los cuales el contribuyente lo realiza de forma manual y en ocasiones los omiten por desconocimiento.

En este sentido la presente tesis pone a disposición un sistema informático que integra todos los procedimientos establecidos en la legislatura ecuatoriana para la declaración del Impuesto a la Renta de Personas Naturales no obligadas a llevar contabilidad.



## 1.2. PLANTEAMIENTO DEL PROBLEMA

**¿Existe un software para determinar el Impuesto a la Renta de personas naturales no obligadas a llevar contabilidad?**

El Impuesto a la Renta es uno de los impuestos mayormente utilizados en varios países del mundo, particular que también se presenta en nuestro país. La importancia del Impuesto a

la Renta radica en sus niveles de recaudación, en el caso del Ecuador, los ingresos tributarios recaudados por el SRI representan un porcentaje muy significativo en el total de ingresos del Presupuesto General del Estado, manteniendo una presión fiscal tal, que obliga a la Administración Tributaria y a los contribuyentes sujetos a este impuesto a que fijen una especial preocupación en su aplicabilidad.

**Cuadro 1**  
**Presupuesto General del Estado**  
**Porcentaje de Fuentes de Ingresos**

FUENTE DE INGRESOS	PORCENTAJE
No Petroleros – TOTAL	75.5
Tributarios – SRI	51.5
No Tributarios	14.7
Tributarios – CAE	9.3
Petroleros – TOTAL	24.5

**Fuente:** SRI 2008

El estudio se concentrará en el cálculo del Impuesto a la Renta de las personas naturales no obligadas a llevar contabilidad, debido a que este sector esta desatendido por los programas informáticos actuales.

Dada la complejidad en el cálculo el Impuesto a la Renta de las personas naturales, es necesario de la presencia de un asesor contable y/o tributario para efectuarlo, labor que sería reemplazada por la herramienta informática propuesta en este proyecto, con el consecuente ahorro de tiempo y dinero, minimizando el riesgo de errores u omisiones



dados por el desconocimiento de la aplicación de la Ley del Régimen Tributario Interno y su reglamento, o por errores involuntarios en el cálculo.





Otro factor importante que debemos tener en cuenta son las reformas insertadas en la Ley de Equidad Tributaria, las cuales complican aún más el cálculo del impuesto, siendo necesario contar con una herramienta informática flexible que permita la incorporación de estos cambios y aquellos que se inserten en el futuro, facilitando al usuario la determinación correcta del impuesto.

## 1.3. SISTEMATIZACIÓN

### 1.3.1. Diagnóstico

El **Servicio de Rentas Internas (SRI)** es una entidad técnica y autónoma que tiene la responsabilidad de recaudar los tributos internos establecidos en la normativa vigente. Su finalidad es la de consolidar la cultura tributaria en el país a efectos de incrementar sostenidamente el cumplimiento voluntario de las obligaciones tributarias por parte de los contribuyentes, su director actual es el Ec. Carlos Marx Carrasco.

Los impuestos que administra y recauda el SRI son

-  Impuesto a la Renta
-  Impuesto al Valor Agregado
-  Impuesto a los Consumos Especiales
-  Impuesto a la Propiedad de Vehículos





## Impuesto a la Salida de Divisas

### **Filosofía**

Fomentar la cultura tributaria en la sociedad ecuatoriana y consolidar la participación de los impuestos administrados por la institución en el financiamiento del Presupuesto General del Estado, mediante la prestación de servicios de calidad al ciudadano a fin de lograr el correcto cumplimiento de sus obligaciones.

El SRI presta servicios a contribuyentes privados y gubernamentales en las 24 provincias del país. Atiende a más de un millón de clientes, mantiene una fuerza laboral de 2.400 empleados y registró un ingreso anual de más de US\$ 5.000 millones en 2007. Entre los servicios que presta se incluyen recepción de declaraciones de impuestos, autorización de comprobantes de venta y manejo de reclamos de los contribuyentes, entre otros. También recauda impuestos y provee información en nombre del gobierno.

### **Objetivos del SRI**

-  Incrementar anualmente la recaudación de impuestos con relación al crecimiento de la economía.
-  Diseñar propuestas de política tributaria orientadas a obtener la mayor equidad, fortalecer la capacidad de gestión institucional y reducir el fraude fiscal.
-  Lograr altos niveles de satisfacción en los servicios al contribuyente.
-  Reducir los índices de evasión tributaria y procurar la disminución en la aplicación de mecanismos de elusión de impuestos.

## Recaudación

Datos estadísticos del Servicio de Rentas Internas demuestran que los ecuatorianos no sólo se preocupan por declarar el impuesto a la renta causado por sus ingresos, sino que exigen facturas de sus compras para poder deducir sus gastos y reducir el valor a pagar. Esto ha minimizado la evasión tributaria e incrementado las recaudaciones del fisco.

**Cuadro 2**  
**Recaudación de Impuestos de Personas naturales 2008**

ESTADÍSTICAS DE RECAUDACIÓN 2008	
Año	2008
Tipo de Contribuyente	PERSONAS NATURALES
Provincia	(Todas)
Cantón	(Todas)
Impuesto a la Renta Recaudado	13.142.971,87
Anticipos al IR	1.482.188,13
Retenciones Mensuales	7.103.354,57
Saldo Anual	4.557.429,17
Impuesto al Valor Agregado	17.485.222,77
Iva de Operaciones Internas	7.983.358,41
Iva Importaciones	9.501.864,36
Impuesto A Los Consumos Especiales	730.450,52
Ice de Operaciones Internas	79.283,02
Ice Importaciones	651.167,50
Impuesto a los Vehículos Motorizados	10.864.879,45
Interés por Mora Tributaria	320.821,68
Multas Tributarias Fiscales	1.031.467,90
Impuesto a la Salida de Divisas	445,36
Régimen Impositivo Simplificado Rise	114.552,01
Otros Ingresos	1.597.408,31

**Fuente:** SRI 2008

Las cifras muestran que en el año 2008, el impuesto a la renta causado incrementó en un 34,9%, en comparación con el año 2007, mientras que en el Impuesto al Valor Agregado, IVA; se logró recaudar el 15,5% en comparación al mismo año.

### Cuadro 3

#### RECAUDACIÓN DEL SERVICIO DE RENTAS INTERNAS FRENTE AL PRESUPUESTO GENERAL DEL ESTADO

Valores en US\$ dólares

GRUPO DE IMPUESTOS	Presupuesto 2008	Recaudación Ene-Dic 2007	Recaudación Ene-Dic 2008	Cumplimiento	Crecimiento 2008 / 2007
<b>TOTAL NETO</b>	<b>5.525.878.180</b>	<b>5.144.110.492</b>	<b>6.194.511.064</b>	<b>112,1%</b>	<b>20,4%</b>
Devoluciones de IVA	(203.000.000)	(217.756.715)	(314.012.825)	-154,7%	44,2%
<b>Impuesto a la Renta Global</b>	<b>1.925.000.000</b>	<b>1.756.774.519</b>	<b>2.369.246.841</b>	<b>123,1%</b>	<b>34,9%</b>
Retenciones Mensuales	1.300.000.000	1.062.960.545	1.413.583.073	108,7%	33,0%
Anticipos al IR	200.000.000	226.738.816	352.325.475	176,2%	55,4%
Saldo Anual	425.000.000	467.075.158	603.338.293	142,0%	29,2%
<b>Impuesto al Valor Agregado</b>	<b>3.167.400.000</b>	<b>3.004.551.505</b>	<b>3.470.518.637</b>	<b>109,6%</b>	<b>15,5%</b>
IVA de Operaciones Internas	1.627.400.000	1.518.385.683	1.762.418.454	108,3%	16,1%
IVA Importaciones	1.540.000.000	1.486.165.823	1.708.100.183	110,9%	14,9%
<b>ICE</b>	<b>497.078.180</b>	<b>456.739.554</b>	<b>473.903.014</b>	<b>95,3%</b>	<b>3,8%</b>
ICE de Operaciones Internas	420.078.180	380.773.224	334.117.559	79,5%	-12,3%
ICE de Importaciones	77.000.000	75.966.330	139.785.455	181,5%	84,0%
Impuesto a los Vehículos	66.000.000	74.356.038	139.785.455	211,8%	88,0%
Intereses por Mora Tributaria	18.000.000	18.253.062	20.402.660	113,3%	11,8%
Multas Tributarias Fiscales	35.400.000	32.807.891	32.191.835	90,9%	-1,9%
Salida de Divisas			31.408.606		
RISE			396.255		
Otros Ingresos	20.000.000	18.384.638	15.139.780	75,7%	-17,6%

Fuente: SRI 2008

## **Situación Actual**

La Administración Tributaria, como parte de su fortalecimiento institucional empezó a realizar continuas campañas fiscalizadoras a las declaraciones de empresas y personas naturales, a partir del año 2000, las mismas que se intensificaron en los últimos años.

Las personas naturales fueron objeto de estas fiscalizaciones y en la mayoría de los casos no se encontraban preparadas, con conocimientos sólidos que les permitieran defender su accionar, realizando en ciertos casos pagos en demasías.

Los contribuyentes que presentan sus declaraciones fuera de los plazos previstos en el Reglamento, deben calcular e incluir en sus declaraciones los intereses y las multas correspondientes.

La multa se calcula cuando el contribuyente no ha presentado la declaración del Impuesto a la Renta, en tanto que el interés por mora se calcula cuando el contribuyente no ha pagado el valor del Impuesto a la Renta cuyo interés anual equivale al 1,5 veces de la tasa activa referencial para noventa días establecida por el Banco Central del Ecuador.

Los contribuyentes generalmente no preparan el registro de ingresos y egresos, incumpliendo la obligación de llevar este registro de acuerdo a lo dispuesto en el Reglamento de Aplicación a la Ley Orgánica de Régimen Tributario Interno.

Por otro lado el Código, la Ley y el Reglamento del Régimen Tributario Interno establecen que los contribuyentes son objeto de retenciones que constituyen un pago anticipado del impuesto a la Renta así como la declaración de 2 anticipos que se realizan en los meses de julio y septiembre por los cuales en ciertos casos se producen pagos indebidos.

**El pago indebido**, de acuerdo al *Art. 122 del Código Tributario* es aquel que se genera cuando el contribuyente no causó impuesto y mantiene un crédito tributario por retenciones en la fuente o anticipos. También se configura cuando un contribuyente causa un impuesto sin que la ley haya establecido alguno o ha pagado de forma duplicada.

**El pago en exceso** se establece en el *Art. 123 del Código Tributario*, como aquel que se genera cuando el contribuyente causó impuesto y mantiene un crédito tributario por retenciones en la fuente o anticipos, que es superior al impuesto causado. También se configura cuando un contribuyente causa un impuesto y pagó un valor mayor al que debió pagar. El contribuyente al desconocer esta situación ha recaído en el pago en demasía de este impuesto sin que se ejerza el derecho a su devolución.

En el siguiente cuadro se puede observar un resumen del resultado obtenido de la encuesta realizada a personas naturales, en la cual se identifican los principales problemas que tienen para declarar el Impuesto a la Renta.

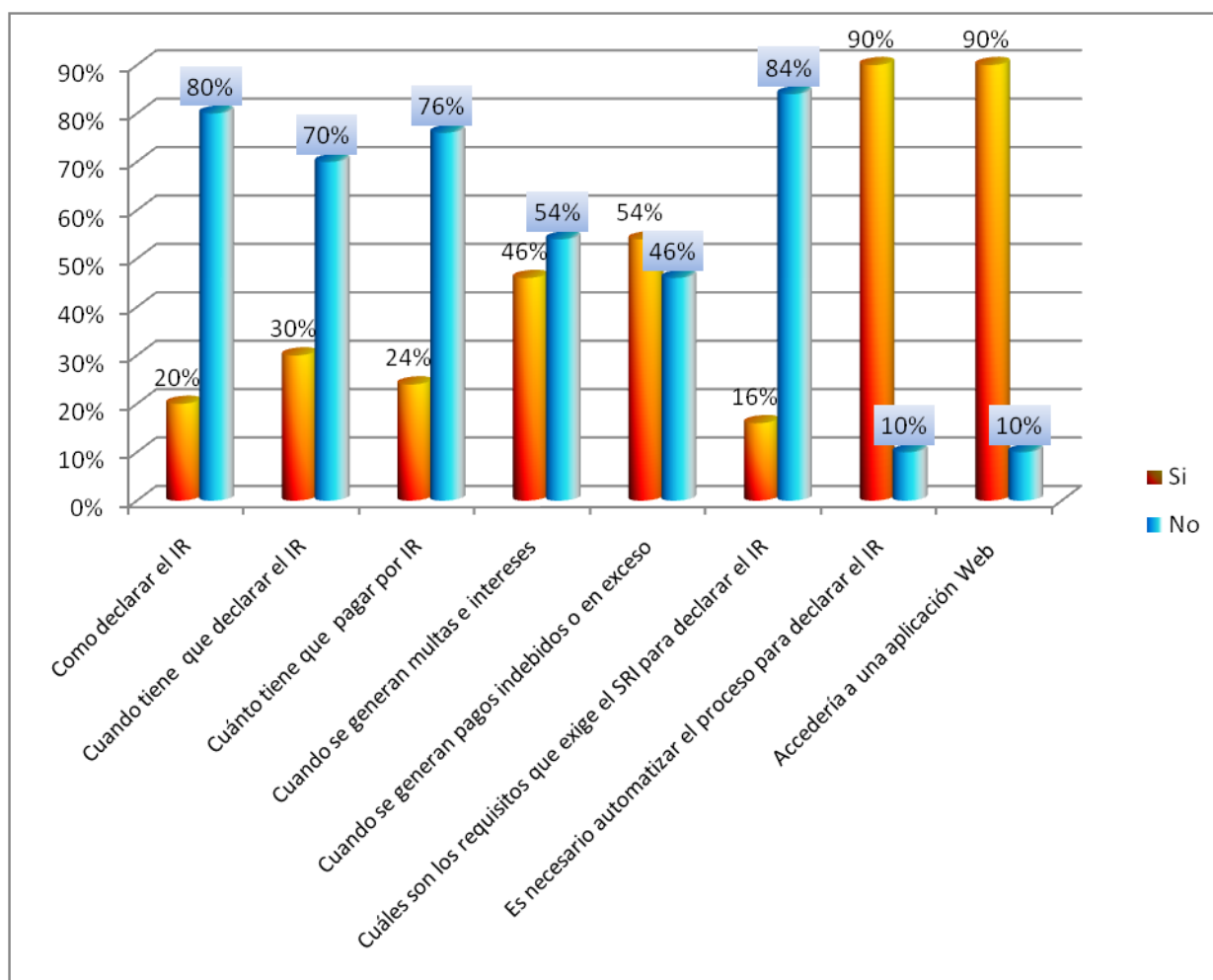
**Cuadro 4**  
**RESUMEN DE PROBLEMAS**

CUADRO RESUMEN - ENCUESTA REALIZADA PARA IDENTIFICACIÓN DE PROBLEMAS	RESPUESTAS				
PREGUNTAS	SI	%	NO	%	TOTAL
¿Conoce usted si existe un sistema para declarar el Impuesto a la Renta?	4	8%	46	92%	50
¿Conoce usted como declarar el Impuesto a la Renta?	10	20%	40	80%	50
¿Conoce usted cuando tiene que declarar el Impuesto a la Renta?	15	30%	35	70%	50
¿Conoce usted cuánto tiene que pagar por Impuesto a la Renta?	12	24%	38	76%	50
¿Conoce usted cuando se generan multas e intereses?	23	46%	27	54%	50
¿Conoce usted cuando se generan pagos indebidos o en exceso?	27	54%	23	46%	50
¿Conoce usted cuáles son los requisitos que exige el SRI para declarar el Impuestos a la Renta?	8	16%	42	84%	50
¿Cree que es necesario automatizar el proceso para declarar el Impuestos a Renta?	45	90%	5	10%	50
¿Cree que es una buena opción crear una aplicación Web a la que se pueda acceder desde cualquier lugar?	45	90%	5	10%	50

**Fuente:** Rafael Chumbi

Gráfico 1

## NIVEL DE CONOCIMIENTO SOBRE IMPUESTO A LA RENTA



**Fuente:** Rafael Chumbi

El 80% de los encuestados manifestó que desconocían como declarar el IR, el 70% desconoce cuando tiene que pagar el impuesto, el 76% no sabe cuánto tiene que pagar por IR, el 46% está consciente de la existencia de multas y pago de intereses en declaraciones tardías, el 54% conoce cuando se generan pagos indebidos o en exceso, el 84% desconoce los requisitos para declarar y el 90% indica que es necesario automatizar el proceso y que accedería a la aplicación en la Web.



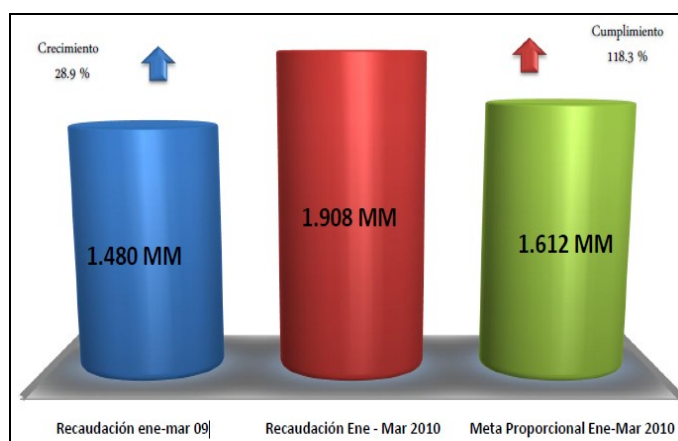
### 1.3.2. Pronóstico

En Ecuador está creciendo considerablemente la cultura tributaria, tras la reforma que realizó el Gobierno en el año 2007.

Datos estadísticos del Servicio de Rentas Internas demuestran que los ecuatorianos no sólo se preocupan por declarar el impuesto a la renta causado por sus ingresos, sino que exigen facturas de sus compras para poder deducir sus gastos y reducir el valor a pagar. Esto ha minimizado la evasión tributaria e incrementado las recaudaciones del fisco.

**Gráfico 2**

#### **RECAUDACIÓN TRIBUTARIA ENERO-MARZO 2010**



**Fuente:** SRI

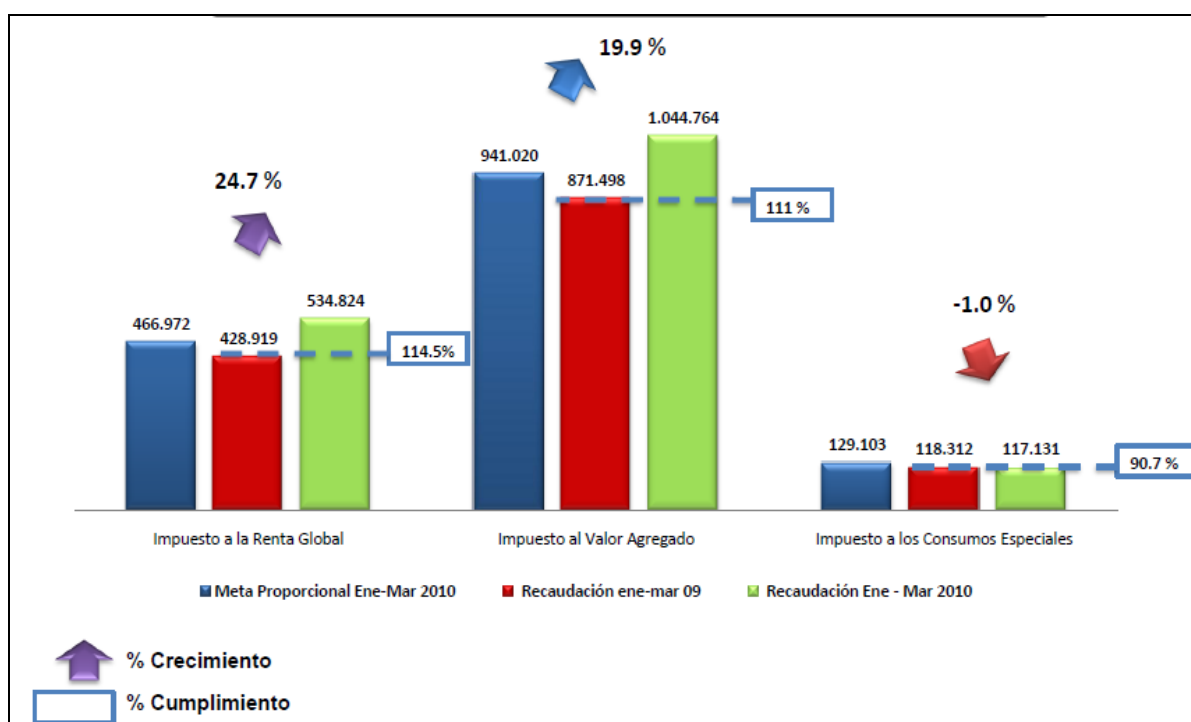
La recaudación enero - marzo 2010 fue de \$1.908 millones de dólares, con un crecimiento nominal del 28.9% frente al mismo período del año anterior. El monto recaudado tuvo un cumplimiento del 118.3% frente a la meta proporcional prevista para el período, constituyendo un aporte adicional de \$295.4 millones de dólares. De los cuales US\$104.8 millones de dólares correspondieron a los Ingresos Extraordinarios, US\$59.3 millones por

Retenciones a la Fuente efectuadas a las exportaciones de petróleo y contratos petroleros por prestaciones de servicios y US\$25.3 millones por el cambio en el pago del impuesto a los vehículos motorizados.

El crecimiento de la recaudación de los Impuestos más relevantes frente a la meta enero – marzo 2010 fueron: Impuesto a la Renta 24.7%, Impuesto al Valor Agregado 19.9%; y, el Impuesto a los Consumos especiales con (-1.0%).

**Gráfico 3**

**CRECIMIENTO NOMINAL DE RECAUDACION ENE – MAR 2010**



Fuente: SRI

La no declaración de impuestos constituye una falta reglamentaria que puede ser sancionada de forma pecuniaria por un valor que oscila de \$30 a \$1500, con la suspensión temporal del RUC o la clausura del negocio.

Cuando el contribuyente persona natural no presenta la declaración del Impuesto a la Renta en los plazos previstos en el reglamento debe calcular e incluir en sus declaraciones una multa de la siguiente manera:

**Cuadro 5**  
**Cuadro para el cálculo de multas**

<b>INFRACCIONES</b>	<b>IMPUESTO A LA RENTA ANUAL</b>
<b>Si causa Impuesto</b>	3% del Impuesto causado por mes o fracción de mes, con un límite máximo del 100% del impuesto causado.
<b>Si no causa Impuesto</b>	0,1% sobre los ingresos brutos obtenidos en el año, por mes o fracción de mes, sin que el valor de la multa exceda del 5% de dichos ingresos

**Fuente:** SRI

El Interés por mora: se calcula esta sanción cuando el contribuyente no ha pagado el valor del Impuesto a la Renta en los plazos previstos en el Reglamento de acuerdo al siguiente cuadro:

## Cuadro 6

### Servicio de Rentas Internas Tasas de Interés Trimestrales por Mora Tributaria

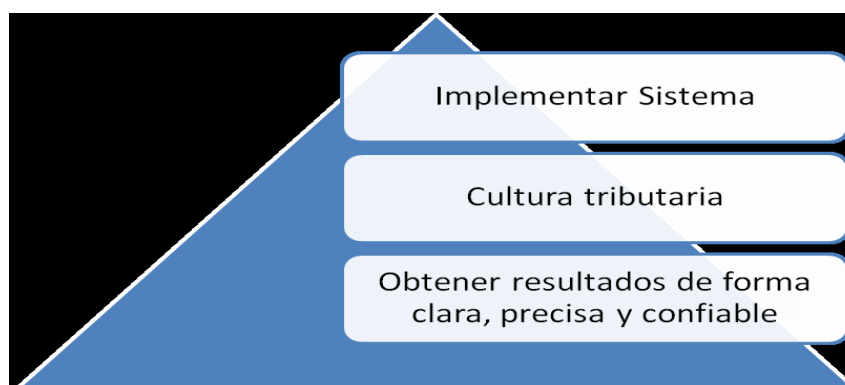
TRIMESTRE	2003	2004	2005	2006	2007	2008	2009	2010	2011
Enero - Marzo	1,171	1,026	0,736	0,824	0,904	1,340	1,143	1,149	1,085
Abril - Junio	1,089	0,994	0,825	0,816	0,860	1,304	1,155	1,151	1,081
Julio - Septiembre	1,115	0,931	0,814	0,780	0,824	1,199	1,155	1,128	1,046
Octubre - Diciembre	1,035	0,885	0,712	0,804	0,992	1,164	1,144	1,130	

Fuente: SRI 2011

### Los pagos indebidos o en exceso:

En el caso de que las retenciones que le hubieren efectuado o los anticipos sean mayores al impuesto causado, el contribuyente podrá solicitar la devolución del impuesto pagado indebidamente o en exceso, presentando una solicitud al Servicio de Rentas Internas, acompañada de la firma de un abogado en el caso del pago indebido ya que el pago en exceso no es necesario del aval del mismo.

### 1.3.3. Control del Pronóstico



Implementar un sistema de ingresos y egresos que permita determinar el Impuesto a la Renta para personas naturales no obligadas a llevar contabilidad atendiendo al art 35 del Reglamento de Aplicación a la Ley Orgánica de Régimen Tributario Interno.

Este registro de ingresos y egresos debe contener la siguiente información: Fecha, comprobante N°, concepto, valor, IVA y observaciones.

El período para declarar se inicia el 1° de febrero y se extiende hasta el mes de marzo de cada año, siendo el sistema quien determine la fecha de pago de acuerdo a la siguiente tabla:

**Cuadro 7**  
**FECHAS DE PAGO IR**

<b>Noveno Dígito</b>	<b>Personas Naturales</b>	<b>Sociedades</b>
<b>1</b>	10 de marzo	10 de abril
<b>2</b>	12 de marzo	12 de abril
<b>3</b>	14 de marzo	14 de abril
<b>4</b>	16 de marzo	16 de abril
<b>5</b>	18 de marzo	18 de abril
<b>6</b>	20 de marzo	20 de abril
<b>7</b>	22 de marzo	22 de abril
<b>8</b>	24 de marzo	24 de abril
<b>9</b>	26 de marzo	26 de abril
<b>0</b>	28 de marzo	28 de abril

**Fuente:** SRI

Al declarar dentro de los plazos previstos en el reglamento, mediante la ayuda de esta herramienta informática se evitará pagar multas e interés por mora tributaria, además que el software al cumplir con las Leyes y el Reglamentos establecidos por el Estado Ecuatoriano evitará pagar valores en demasía.

Con la utilización de este software el contribuyente estaría en la posibilidad de determinar el valor del impuesto a pagar o en su defecto el saldo del crédito tributario a favor pudiendo el usuario proceder con el trámite de devolución del mismo sea por pago en exceso o indebido.

## **1.4. OBJETIVOS**

### **1.4.1. Objetivo general**

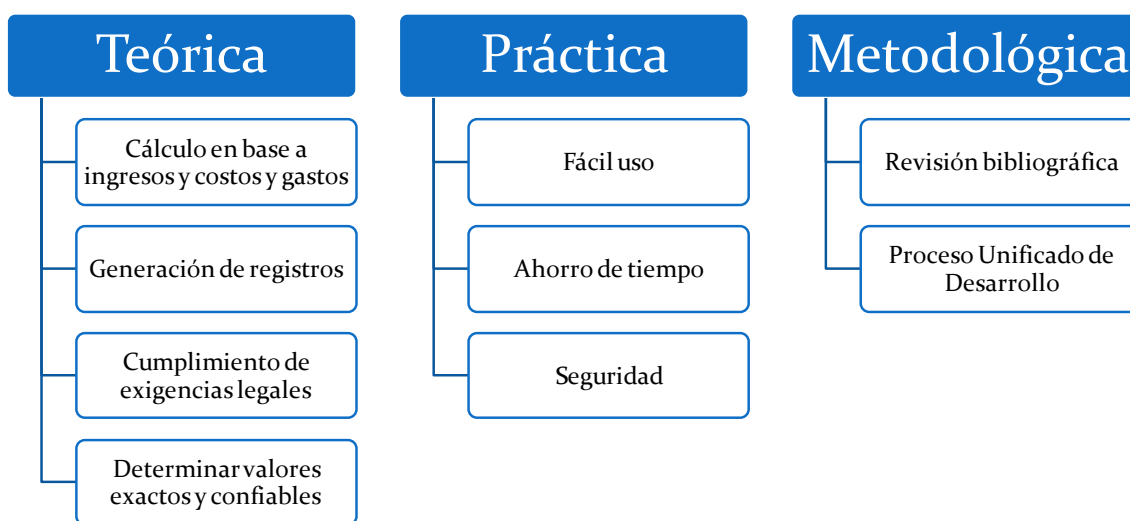
Desarrollar e implementar un sistema de ingresos y egresos para determinar el impuesto a la renta en la web para personas naturales no obligadas a llevar contabilidad

### **1.4.2. Objetivos específicos**

- ✚ Analizar el flujo de la información necesaria para determinar el Impuesto a la Renta de personas naturales no obligadas a llevar contabilidad.
- ✚ Aplicar la metodología orientada a objetos para automatizar el cálculo del Impuesto a la Renta de personas naturales no obligadas a llevar contabilidad.

- ✚ Desarrollar registros digitales de Ingresos y Egresos de cada usuario.
- ✚ Determinar el impuesto causado, valor a pagar, anticipos, multas e intereses según corresponda para cada usuario, garantizando la integridad y confiabilidad de los datos.
- ✚ Diseñar la Página Web para el sistema de ingresos y egresos para determinar el impuesto a la renta para personas naturales no obligadas a llevar contabilidad

## 1.5. JUSTIFICACIÓN



### 1.5.1. Justificación Teórica

El impuesto a la Renta se debe cancelar sobre los ingresos o rentas, producto de actividades personales, comerciales, industriales, agrícolas, y en general actividades económicas y aún sobre ingresos gratuitos, percibidos durante un año, luego de descontar los costos y gastos incurridos para obtener o conservar dichas rentas.

En la actualidad no existe un software que permita a las personas naturales tener un registro tanto de ingresos como de egresos para el cálculo exacto de dicho impuesto, por lo que con la implementación de este software, se determinara el Impuesto a la Renta bajo las exigencias de las leyes ecuatorianas, obteniendo valores exactos y confiables para la declaración del Impuesto a la Renta de personas naturales no obligadas a llevar contabilidad.

El tema propuesto busca, mediante Interfaces gráficas (graphical user interfaces, GUIs) proporcionar al usuario las herramientas necesarias para lograr determinar el Impuesto a la Renta y representar de forma gráfica en la pantalla del ordenador los datos y objetos, obteniendo las siguientes ventajas:

- ✚ Ver en la pantalla los gráficos y textos tal como se verán impresos.
- ✚ Manipular en la pantalla directamente los objetos y la información.
- ✚ Elementos de interfaz estándar como menús y diálogos.
- ✚ Proporciona flexibilidad en el uso de dispositivos de entrada (teclado/ratón).

Este sistema de ingresos y egresos al ser una aplicación Web contiene elementos que permiten una comunicación activa entre el usuario y la información; esto permite que el usuario acceda a los datos de modo interactivo, gracias a que la página responderá a cada una de sus acciones, como por ejemplo rellenar formularios.



### 1.5.2. Justificación Práctica

Los resultados del SISTEMA DE INGRESOS Y EGRESOS PARA DETERMINAR EL IMPUESTO A LA RENTA EN LA WEB PARA PERSONAS NATURALES NO OBLIGADAS A LLEVAR CONTABILIDAD, serán medidos en cuanto a: fácil uso, ahorro de tiempo y seguridad.

**Fácil uso:** por medio de la interfaz gráfica (Graphical User Interfaces, GUIs) el usuario interactúa y establece un contacto más fácil e intuitivo con el sistema, permitiendo el ingreso de datos de forma sencilla utilizando pantallas que orientan o sugieren la información que debe ser ingresada.

**Ahorro de tiempo:** La utilización del SISTEMA DE INGRESOS Y EGRESOS PARA DETERMINAR EL IMPUESTO A LA RENTA EN LA WEB PARA PERSONAS NATURALES NO OBLIGADAS A LLEVAR CONTABILIDAD, en lugar de los sistemas tradicionales, en los que todas las operaciones se realizan de forma manual, ahorrará gran cantidad de tiempo, esto es posible, debido a que, en el sistema ya están incorporados valores, fórmulas, tablas y constantes que intervienen para el cálculo del Impuesto a la Renta.

**Seguridad:** Al contar el sistema con un amplio componente legal desde el punto de vista tributario, tenemos la seguridad de que el cálculo del impuesto cumple con las exigencias legales evitando que tengamos inconvenientes con las instituciones gubernamentales. Además al automatizar el cálculo para determinar el impuesto se eliminan los errores manuales garantizando un resultado confiable.

### **1.5.3. Justificación Metodológica**

Para desarrollar este software se utilizará el Proceso Unificado de Desarrollo.

El ciclo de vida RUP es una implementación del Desarrollo en espiral. El ciclo de vida organiza las tareas en fases e iteraciones.

RUP divide el proceso en cuatro fases, dentro de las cuales se realizan varias iteraciones en número variable según el proyecto y en las que se hace un mayor o menor hincapié en las distintas actividades.

Durante la fase de inicio las iteraciones hacen mayor énfasis en actividades de modelado del negocio y de requerimientos.

En la fase de elaboración, las iteraciones se orientan al desarrollo de la baseline de la arquitectura, abarcan más los flujos de trabajo de requerimientos, modelo de negocios (refinamiento), análisis, diseño y una parte de implementación orientado a la baseline de la arquitectura.

En la fase de construcción, se lleva a cabo la construcción del producto por medio de una serie de iteraciones.

En la fase de transición se pretende garantizar que se tiene un producto preparado para su entrega a la comunidad de usuarios.

Como se puede observar en cada fase participan todas las disciplinas, que dependiendo de la fase, se concentra el esfuerzo más o menos en determinada disciplina.

## 1.6. ALCANCE

Esta investigación propone desarrollar un software orientado a la Web que permita determinar el Impuesto a la Renta de una persona natural, que realicen actividad empresarial y que operen con un capital propio inferior a USD 60.000, o sus ingresos brutos anuales sean inferiores a \$100.000, o sus costos y gastos no superen los USD 80.000; así como los profesionales, agentes o comisionistas, trabajadores autónomos y artesanos.

El software estará estructurado principalmente de los siguientes módulos:

**Sistema de Ingresos y egresos:** De acuerdo al art 35 del Reglamento de Aplicación a la Ley Orgánica de Régimen Tributario Interno existe la obligación de llevar un registro de ingresos y egresos que debe contener la siguiente información:

- ✚ Fecha
- ✚ Comprobante N°
- ✚ Concepto
- ✚ Valor
- ✚ IVA
- ✚ Observaciones

**Gastos personales:** El Art. 31 de la Ley Orgánica de Régimen establece que: las personas naturales podrán deducirse sus gastos personales, sin IVA ni ICE, así como los de su cónyuge o conviviente e hijos menores de edad o con discapacidad, que no perciban ingresos gravados y que dependan del contribuyente.

Los gastos personales que se pueden deducir, corresponden a los realizados por concepto de: vivienda, educación, salud, alimentación y vestimenta.

**Determinar el Impuesto:** para declarar el impuesto causado se utilizará el formulario 102A, que debe cumplir con los siguientes requisitos:

- ✚ Los apellidos y nombres del contribuyente y el número del RUC o cédula de Ciudadanía que constan en el formulario 102A, deben coincidir con el documento de inscripción del RUC o cédula que se presentará al momento de realizar la declaración, en caso de que ésta se realice a través del sistema financiero.
- ✚ Identifique correctamente y en forma clara el ejercicio fiscal declarado, para el presente caso en la casilla correspondiente del formulario debe registrar “2009”.
- ✚ Todos los casilleros no utilizados deben ser anulados con una línea horizontal, para anular una casilla no debe utilizar ceros.
- ✚ El formulario de declaración debe ser llenado a máquina (de escribir o impresora) o, con letras de imprenta cuando se lo hace manualmente. No se puede utilizar lápiz para elaborar la declaración.
- ✚ Tanto el original como las copias no deben contener manchones, repisados, enmendaduras o escritos que no correspondan a los casilleros del formulario.
- ✚ Utilizar el punto para separar miles y la coma para decimales; siempre se deben utilizar dos decimales, aproximando el segundo.

- ✚ Presentar un solo formulario de declaración por cada periodo fiscal.
- ✚ La declaración debe consolidar la información de todas las actividades y establecimientos que correspondan a un mismo contribuyente.
- ✚ Los formularios utilizan papel químico que permite obtener dos copias.

## **1.7. ESTUDIOS DE FACTIBILIDAD**

### **1.7.1. Factibilidad Técnica**

Para el desarrollo del software se utilizarán las siguientes herramientas de desarrollo:

#### **Framework de desarrollo**

**.NET** es un proyecto de Microsoft para crear una nueva plataforma de desarrollo de software con énfasis en transparencia de redes, con independencia de plataforma de hardware y que permite un rápido desarrollo de aplicaciones. Basado en ella, la empresa intenta desarrollar una estrategia horizontal que integre todos sus productos, desde el sistema operativo hasta las herramientas de mercado.

**.NET** podría considerarse una respuesta de Microsoft al creciente mercado de los negocios en entornos Web, como competencia a la plataforma Java de Sun Microsystems y a los diversos framework de desarrollo web basados en PHP.

Su propuesta es ofrecer una manera rápida y económica, a la vez que segura y robusta, de desarrollar aplicaciones permitiendo una integración más rápida y ágil entre empresas y un acceso más simple y universal a todo tipo de información desde cualquier tipo de dispositivo.

Los principales componentes del marco de trabajo son:

- ✚ El conjunto de lenguajes de programación

- ✚ La Biblioteca de Clases Base o BCL

El **Entorno Común de Ejecución para Lenguajes** o CLR por sus siglas en inglés.

Debido a la publicación de la norma para la **infraestructura común de lenguajes** (CLI), el desarrollo de lenguajes se facilita, por lo que el **marco de trabajo .NET** soporta ya más de 20 lenguajes de programación y es posible desarrollar cualquiera de los tipos de aplicaciones soportados en la plataforma con cualquiera de ellos, lo que elimina las diferencias que existían entre lo que era posible hacer con uno u otro lenguaje.

Se utilizara Visual Web Developer 2005 Express Edition ya que es la versión gratuita que Microsoft ofrece para desarrolladores de software.

**Cuadro 8**  
**Matriz de prioridades: Plataforma de desarrollo**

CARACTERÍSTICAS	Orientado a Objetos	Fiabilidad	Facilidad de conexión	Generación de aplicaciones Web	Portable	Independiente a la arquitectura	Creación de aplicaciones	Seguridades	Total	Porcentaje
Orientado a Objetos	1	1	0,5	1	0,3	1	0,5	6,3	16%	
Fiabilidad	1	0	0	0,5	0,3	0	1	3,3	8%	
Facilidad de conexión	1	0	0,5	1	0,5	0,5	0,5	4,3	11%	
Generación de aplicaciones Web	0,5	0	0,5	0,5	0,8	1	0,4	4,7	12%	
Portable	1	0,5	1	0,5	0,5	0,5	0,5	5	12%	
Independiente a la arquitectura	0,3	0,3	0,5	0,8	0,5	0,5	0,5	3,7	9%	
Creación de aplicaciones Distribuidas	1	0	0,5	1	0,5	0,5	0	4	10%	
Seguridades	0,5	1	0,5	0,4	0,5	0,5	0	4,4	11%	
<b>Total</b>								<b>40,3</b>	<b>100%</b>	

**Fuente:** Rafael Chumbi

**Cuadro 9**  
**Matriz de Ponderación: Plataforma de desarrollo**

Características	Plataforma	PONDERACIÓN	JAVA		.NET	
			valor	total	valor	Total
Orientado a Objetos		16%	4	0,6	5	0,8
Fiabilidad		8%	4	0,3	5	0,4
Facilidad de Conexión		11%	3	0,3	4	0,4
Generación de Aplicaciones Web		12%	5	0,6	5	0,6
Portable		12%	5	0,6	5	0,6
Independiente a la Arquitectura		9%	5	0,4	5	0,4
Creación de aplicaciones Distribuidas		10%	5	0,5	5	0,5
Seguridades		11%	4	0,4	4	0,4
<b>Total</b>		<b>100%</b>		<b>4,1</b>		<b>4,6</b>

**Fuente:** Rafael Chumbi

## Base de datos

### Microsoft SQL Server 2005 Express

SQL Server 2005 introduce la posibilidad de desarrollar objetos de base de datos en lenguajes .NET. Pueden crearse Objetos de código, incluyendo Funciones, Procedimientos y Triggers en lenguajes como C# y VB.NET. Permite servicios Web en la capa de base de datos, convirtiendo a SQL Server en un servidor HTTP. Esta capacidad supone un nuevo modelo de acceso a los datos para aplicaciones basadas en Web.

**Cuadro10**

### Matriz de prioridades: Base de datos

Características	Open Source	Orientada a Objetos	de Velocidad Respuesta	Cliente/Servidor	Integridad referencial	Escalabilidad	Total	Porcentaje
Open Source	1	1	0,5	1	0,5	5	21,7%	
Orientada a Objetos	1	0,5	0,5	0,5	0,5	3,5	15,21%	
Velocidad de Respuesta	1	0,5	1	0,5	0,5	4,5	19,56%	
Cliente/Servidor	0,5	0,5	1	0,5	0,5	3,5	15,21%	
Integridad referencial	1	0,5	0,5	0,5	0,5	3,5	15,21%	
Escalabilidad	0,5	0,5	0,5	0,5	0,5	3	13,04%	
<b>Total</b>						<b>23</b>	<b>100%</b>	

**Fuente:** Rafael Chumbi



### Cuadro11

#### Matriz de Ponderación: Base de datos

Características	Ponderación	ORACLE 8i		SQL SERVER 2000 EXPRESS	
		Valor	Total	Valor	Total
Open Source	21,7%	0	0	5	1
Orientada a Objetos	15,21%	5	0,76	5	0,76
Velocidad de Respuesta	19,56%	5	0,9	4	0,76
Cliente/Servidor	15,21%	5	0,76	5	0,76
Integridad referencial	15,21%	5	0,76	4	0,6
Escalabilidad	13,04%	5	0,65	4	0,5
<b>Total</b>	<b>100%</b>		<b>4,0</b>		<b>4,4</b>

**Fuente:** Rafael Chumbi

#### Proceso de ciclo de vida

Proceso unificado de desarrollo, con este ciclo de vida me permite mejorar el producto en cada interacción obteniendo al final un software que cumple con todos los requisitos y por lo tanto de alta calidad.

**Cuadro 12**  
**Matriz de Prioridades: Ciclo de vida**

CARACTERÍSTICAS	Rapidez en Desarrollo	Interacción usuario	Impacto	Personal Involucrado	Mayor gestión de complejidad	Fiabilidad	Visibilidad	Componentes Orientados a	Total	Porcentaje
Rapidez en Desarrollo	1	1	1	0,5	1	1	1	6,5	19%	
Interacción usuario	1	0,5	0,5	0	0	0,5	0,5	3	9%	
Impacto	1	0,5	1	0,5	0,5	0,5	0,5	4,5	13%	
Personal Involucrado	1	0,5	1	1	0,5	0,5	1	5,5	16%	
Mayor gestión de complejidad	0,5	0	0,5	1	0,5	0	0,5	3	9%	
Fiabilidad	1	0	0,5	0,5	0,5	0,5	0,5	3,5	10%	
Visibilidad	1	0,5	0,5	0,5	0	0,5	0,5	3,5	10%	
Componentes Orientados a Objetos	1	0,5	0,5	1	0,5	0,5	0,5	4,5	13%	
<b>Total</b>								<b>34</b>	<b>100%</b>	

**Fuente:** Rafael Chumbi

**Cuadro13**  
**Matriz de Ponderación: Ciclo de vida**

Características	Ponderación	Línea Secuencial		Construcción Prototipos		Espiral		RUP	
		valor	total	valor	total	valor	total	valor	total
Rapidez en Desarrollo	19%	5	1	4	0.8	4	0.8	4	0.8
Interacción usuario	9%	4	0.4	4	0.4	5	0.5	5	0.5
Impacto	13%	1	0.1	5	0.7	4	0.5	4	0.5
Personal Involucrado	16%	3	0.5	3	0.5	5	0.8	5	0.8
Mayor gestión de complejidad	9%	4	0.4	5	0.5	5	0.5	5	0.5
Fiabilidad	10%	4	0.40	4	0.4	5	0.50	5	0.50
Visibilidad	10%	3	0.30	3	0.3	4	0.40	4	0.40
Componentes Orientados a Objetos	13%	4	0.4	3	0.4	5	0,7	5	0,7
<b>Total</b>	<b>100%</b>		<b>3.5</b>		<b>4.0</b>		<b>4.70</b>		<b>4.70</b>

**Fuente:** Rafael Chumbi

## Equipos

Para el desarrollo del sistema se cuenta con equipos que contienen las siguientes características:

- ✚ Microsoft Visual Web Developer 2005 Express Edition
- ✚ Microsoft SQL Server Management Studio Express.
- ✚ Microsoft Office 2007.
- ✚ Microsoft Internet Explorer.
- ✚ Windows Xp Service Pack II

**Cuadro 14**

### Hardware del servidor

<b>Procesador</b>	1 Procesador Intel IV de 2GHZ
<b>Memoria</b>	1 GB
<b>Almacenamiento</b>	160 GB
<b>Monitor</b>	17 pulgadas
<b>Red</b>	Tarjeta de red Fast Ethernet 10 base 1000
<b>MODEM</b>	Tarjeta fax MODEM

**Fuente:** Rafael Chumbi

**Cuadro 15****Matriz de prioridades: Sistema Operativo**

CARACTERÍSTICAS	Costo	Estabilidad	Facilidad	Seguridad	Multitarea	Compatibilidad con otros sistemas	Total	Porcentaje
Costo	1	1	1	1	1	1	5	20%
Estabilidad	1	0,5	1	0,5	1	1	4	16%
Facilidad	1	1	0,5	1	1	1	4,5	18%
Seguridad	1	1	0,5	0,5	0,5	0,5	3,5	14%
Multitarea	1	0,5	1	0,5	0,5	0,5	3,5	14%
Compatibilidad con otros sistemas	1	1	1	0,5	0,5	0,5	4	16%
<b>Total</b>							<b>24,5</b>	<b>100%</b>

Fuente: Rafael Chumbi

**Cuadro 16****Matriz de Ponderación: Sistema Operativo**

Características	Ponderación	WINDOWS XP		WINDOWS 2000	
		valor	total	valor	Total
Costo	20%	3	0.6	1	0.2
Estabilidad	16%	4	0.7	4	0.7
Facilidad	18%	4	0.7	4	0.7
Seguridad	14%	3	0.4	4	0.6
Multitarea	14%	4	0.6	1	0.1
Compatibilidad con otros sistemas	16%	5	0.8	5	0.8
<b>Total</b>	<b>100%</b>		<b>3.8</b>		<b>3.1</b>

Fuente: Rafael Chumbi

El sistema Operativo será Windows XP, ya que la mayoría de usuarios están familiarizados con este sistema.

### **1.7.2. Factibilidad Económica**

Para medir la factibilidad económica del proyecto se procedió a analizar la relación costo beneficio del mismo incluyendo los ingresos que se recibirían al colocar el producto en un mercado de 50 clientes, considerando gastos básicos como: compra de licencia del software, el valor de horas trabajadas, suministros de oficinas e impresión, así como el gasto de publicidad y promoción del producto, obteniendo una rentabilidad del 33%, valor que supera la inflación existente en el país, así como también la tasa máxima de interés que pagan los bancos en la actualidad.

Además un punto importante a considerar es que el costo de mano de obra es un beneficio adicional para el programador, por lo cual la rentabilidad es aún mayor.

Como se puede observar en el cuadro adjunto,  $B/C > 1$ , dando una relación de 1.33, lo cual implica que los ingresos son mayores que los egresos, demostrando que el proyecto es factible.

Cuadro 17

## RELACIÓN COSTO BENEFICIO

	Cantidad	Valor	Total
<b>Beneficios</b>			
Capacitación	50	30	1500
Venta Software	50	50	2500
<b>Total Beneficios</b>			<b>4000</b>
<b>Costos</b>			
Horas Hombre	360	8	2880
Suministros de Oficina	6	3	18
Impresión	1	40	40
Publicidad	2	40	80
<b>Total Costos</b>			<b>3018</b>
<b>Relación Costo Beneficio</b>			
<b>Beneficio/Costo=</b>			<b>1,33</b>

**Fuente:** Rafael Chumbi

### 1.7.3. Factibilidad Operacional

Con el análisis de esta factibilidad operacional podemos determinar la probabilidad de que el sistema se use de manera adecuada.

El sistema es de fácil uso y entendimiento para los usuarios, reduciendo el riesgo de que el usuario incurra en errores o fallas en el sistema, además tiene una interfaz amigable para que el usuario no se complique y llegue a ignorarlo.

El sistema va a ser de gran apoyo ya que el cálculo para el pago de impuesto es complejo, por lo tanto con el sistema se ahorrara tiempo para el usuario constituyéndose en una herramienta para su desempeño de tal forma que el usuario se sienta apoyado y no

desplazado por el sistema. La tecnología que va a utilizarse está de acuerdo a las necesidades del usuario y de acuerdo al entorno en el cual va a trabajar, por lo tanto de ninguna manera es tecnología obsoleta ni lo será por algún tiempo.

## 2. MARCO DE REFERENCIA

### 2.1. MARCO TEÓRICO

“Un portal de Internet es un sitio Web cuyo servicio es ofrecer al usuario, de forma fácil e integral, el acceso a una serie de recursos y servicios, entre los que suelen encontrarse buscadores, foros, documentos, aplicaciones, compra electrónica, etc. Principalmente están dirigidos a resolver necesidades específicas de un grupo de personas o de acceso a la información y servicios de una institución pública o privada.”  
([http://es.wikipedia.org/wiki/Portal\\_\(Internet\)](http://es.wikipedia.org/wiki/Portal_(Internet)))

“Como el software, al igual que el capital, es el conocimiento incorporado, y puesto que el conocimiento está inicialmente disperso, el desarrollo del software implícito, latente e incompleto en gran medida, es un proceso social de aprendizaje.

El proceso es un diálogo en el que se reúne el conocimiento y se incluye en el software para convertirse en software. El proceso proporciona una interacción entre los usuarios y los diseñadores, entre los usuarios y las herramientas de desarrollo [tecnología]. Es un proceso interactivo donde la herramienta de desarrollo se usa como medio de comunicación, con cada iteración del diálogo se obtiene mayor conocimiento de las personas involucradas.” (Baetjer, 1998: 13).

La orientación a objetos ha tomado por asalto y en forma legítima al mundo del software. Como medio para la generación de programas, tiene varias ventajas. Fomenta una metodología basada en componentes para el desarrollo de software de manera que primero se genera un sistema mediante un conjunto de objetos, luego se podrá ampliar el sistema



agregándole funcionalidad a los componentes que ya había generado o agregándole nuevos componentes, y finalmente podrá volver a utilizar los objetos que generó para el sistema cuando cree uno nuevo, con lo cual reducirá sustancialmente el tiempo de desarrollo de un sistema.

Un objeto es la instancia de una clase (o categoría). Un objeto cuenta con una estructura, es decir atributos (propiedades) y acciones.

Las acciones son todas las actividades que el objeto es capaz de realizar. Los atributos y acciones, en conjunto, se conocen como características o rasgos.

La orientación a objetos se refiere a algo más que tan solo atributos y acciones; también considera otros aspectos: como abstracción, herencia, polimorfismo y encapsulamiento o encapsulación. Otros factores importantes de la orientación a objetos son: el envío de mensajes, las asociaciones y la agregación. PRESSMAN, Roger S. 1997.

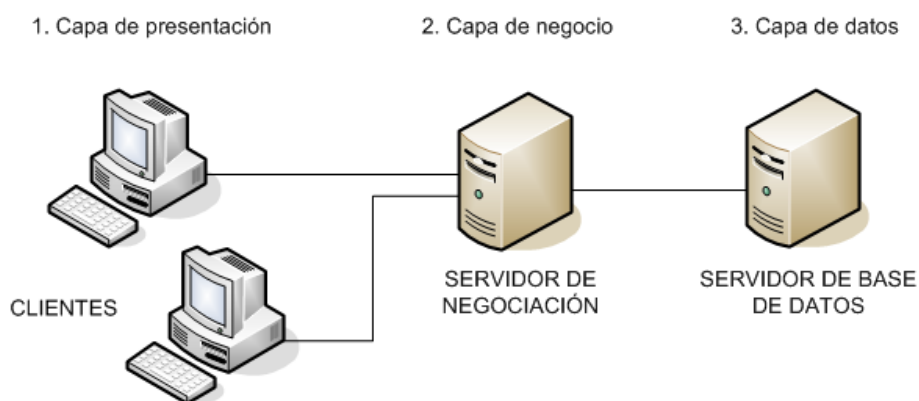
El Sistema de gestión de bases de datos consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a dichos datos.

La colección de datos, normalmente denominada base de datos, contiene información acerca de una empresa particular.

El primer objetivo de un SGBD es proporcionar un entorno tanto práctico como eficiente de usar en la recuperación y el almacenamiento de la información de la base de datos. Los sistemas de bases de datos se diseñan para gestionar grandes cantidades de información; la gestión de los datos implica tanto la definición de estructuras para almacenar la información como la provisión de mecanismos para la manipulación de la misma.

En suma, los sistemas de bases de datos deben proporcionar la fiabilidad de la información almacenada, a pesar de las caídas del sistema o los intentos de acceso sin autorización. Si los datos van a ser compartidos entre diversos usuarios, el sistema debe evitar posibles resultados anómalos.” (Silberschatz. Korth. Sudarshan., 1998: 1)

“La programación por capas es un estilo de programación en el que el objetivo primordial es la separación de la lógica de negocios de la lógica de diseño; un ejemplo básico de esto consiste en separar la capa de datos de la capa de presentación al usuario.



La ventaja principal de este estilo es que el desarrollo se puede llevar a cabo en varios niveles y, en caso de que sobrevenga algún cambio, sólo se ataca al nivel requerido sin tener que revisar entre código mezclado. Un buen ejemplo de este método de programación sería el modelo de interconexión de sistemas abiertos.

Además, permite distribuir el trabajo de creación de una aplicación por niveles; de este modo, cada grupo de trabajo está totalmente abstraído del resto de niveles, de forma que basta con conocer la API que existe entre niveles.

En el diseño de sistemas informáticos actual se suele usar las arquitecturas multinivel o Programación por capas; en dichas arquitecturas a cada nivel se le confía una misión simple, lo que permite el diseño de arquitecturas escalables (que pueden ampliarse con facilidad en caso de que las necesidades aumenten).

El diseño más utilizado actualmente es el diseño en tres niveles (o en tres capas).

**1.- Capa de presentación:** es la que ve el usuario (también se la denomina "capa de usuario"), presenta el sistema al usuario, le comunica la información y captura la información del usuario en un mínimo de proceso (realiza un filtrado previo para comprobar que no hay errores de formato).

Esta capa se comunica únicamente con la capa de negocio también conocida como interfaz gráfica y debe tener la característica de ser "amigable" (entendible y fácil de usar) para el usuario.

**2.- Capa de negocio:** es donde residen los programas que se ejecutan, se reciben las peticiones del usuario y se envían las respuestas tras el proceso. Se denomina capa de negocio (e incluso de lógica del negocio) porque es aquí donde se establecen todas las reglas que deben cumplirse.

Esta capa se comunica con la capa de presentación, para recibir las solicitudes y presentar los resultados, y con la capa de datos, para solicitar al gestor de base de datos el almacenaje o recuperación de datos. También se consideran aquí los programas de aplicación.

**3.- Capa de datos:** es donde residen los datos y es la encargada de acceder a los mismos, está formada por uno o más gestores de bases de datos que realizan todo el almacenamiento de datos, reciben solicitudes de almacenamiento o recuperación de información desde la capa de negocio.

Todas estas capas pueden residir en un único ordenador, si bien lo más usual es que haya una multitud de ordenadores en donde reside la capa de presentación (son los clientes de la arquitectura cliente/servidor).

Las capas de negocio y de datos pueden residir en el mismo ordenador, y si el crecimiento de las necesidades lo aconseja se pueden separar en dos o más ordenadores. Así, si el tamaño o complejidad de la base de datos aumenta, se puede separar en varios ordenadores los cuales recibirán las peticiones del ordenador en el que resida la capa de negocio.

Si, por el contrario, fuese la complejidad en la capa de negocio lo que obligase a la separación, esta capa de negocio podría residir en uno o más ordenadores que realizarían solicitudes a una única base de datos.

En sistemas muy complejos se llega a tener una serie de ordenadores sobre los cuales corre la capa de negocio, y otra serie de ordenadores sobre los cuales corre la base de datos. En una arquitectura de tres niveles, los términos "*capas*" y "*niveles*" no significan lo mismo ni son similares.

El término "*capa*" hace referencia a la forma como una solución es segmentada desde el punto de vista lógico: Presentación/ Lógica de Negocio/ Datos.

En cambio, el término "nivel" corresponde a la forma en que las capas lógicas se encuentran distribuidas de forma física. Por ejemplo: Una solución de tres capas (presentación, lógica del negocio, datos) que residen en un solo ordenador (Presentación+lógica+datos). Se dice que la arquitectura de la solución es de tres capas y un nivel.

Una solución de tres capas (presentación, lógica del negocio y datos) que residen en dos ordenadores (presentación + lógica, lógica + datos). Se dice que la arquitectura de la solución es de tres capas y dos niveles.”

([http://es.wikipedia.org/wiki/Programaci%C3%B3n\\_por\\_capas](http://es.wikipedia.org/wiki/Programaci%C3%B3n_por_capas))

## **2.2. MARCO CONCEPTUAL**

### **Visual Studio .NET**

Visual Studio.NET es un conjunto completo de herramientas para la creación tanto de aplicaciones de escritorio como las de Web para trabajo en equipo.

Aparte de generar aplicaciones de escritorio de alto rendimiento, se pueden utilizar las eficaces herramientas de desarrollo basado en componentes y otras tecnologías de Visual Studio para simplificar el diseño, desarrollo e implementación en equipo de soluciones para la empresa.

Visual Studio es un conjunto de herramientas de desarrollo para la generación de aplicaciones Web ASP.NET, Servicios Web XML, aplicaciones de escritorio y aplicaciones móviles.

Visual Basic, Visual C++, Visual C# y Visual J# utilizan el mismo entorno de desarrollo integrado (IDE), que les permite compartir herramientas y facilita la creación de soluciones en varios lenguajes. Asimismo, dichos lenguajes aprovechan las funciones de **.NET Framework**, que ofrece acceso a tecnologías clave para simplificar el desarrollo de aplicaciones Web ASP y Servicios Web XML.

### **El entorno .NET Framework**

.NET Framework es un entorno multilenguaje que permite generar, implantar y ejecutar aplicaciones y Servicios Web XML. Consta de tres partes principales:

#### **Common Language Runtime**

A pesar de su nombre, el motor en tiempo de ejecución desempeña una función durante la ejecución y durante el desarrollo de los componentes. Cuando el componente se está ejecutando, el motor en tiempo de ejecución es responsable de administrar la asignación de memoria, iniciar y detener subprocessos y procesos, y hacer cumplir la directiva de seguridad, así como satisfacer las posibles dependencias del componente sobre otros componentes.

Durante el desarrollo, el papel del motor en tiempo de ejecución cambia ligeramente; a causa de la gran automatización que permite (por ejemplo, en la administración de memoria), el motor simplifica el trabajo del desarrollador, especialmente al compararlo con la situación actual de la tecnología COM.

En concreto, funciones tales como la reflexión reducen de forma espectacular la cantidad de códigos que debe escribir el desarrollador para convertir la lógica de empresa en componentes reutilizables.

### **Clases de programación unificadas**

El entorno de trabajo ofrece a los desarrolladores un conjunto unificado, orientado a objetos, jerárquico y extensible de bibliotecas de clases (API). Actualmente, los desarrolladores de C++ utilizan la Microsoft Foundation Classes y los desarrolladores de Java utilizan las Windows Foundation Classes.

El entorno de trabajo unifica estos modelos dispares y ofrece a los programadores de Visual Basic y JScript la posibilidad de tener también acceso a las bibliotecas de clases. Con la creación de un conjunto de API comunes para todos los lenguajes de programación, Common Language Runtime permite la herencia, el control de errores y la depuración entre lenguajes. Todos los lenguajes de programación, desde JScript a C++, pueden tener acceso al entorno de trabajo de forma parecida y los desarrolladores pueden elegir libremente el lenguaje que desean utilizar.

### **ASP.NET ASP.NET**

Construye las clases de programación de .NET Framework, lo que proporciona un modelo de aplicación Web con un conjunto de controles e infraestructura que facilitan la generación de aplicaciones Web. ASP.NET incluye un conjunto de controles que

encapsulan elementos comunes de interfaz de usuario de HTML, como cuadros de texto, botones y cuadros de lista.

Sin embargo, los controles se ejecutan en el servidor Web, y representan la interfaz de usuario en el explorador como HTML. En el servidor, los controles exponen un modelo de programación orientado a objetos que proporciona la riqueza de la programación orientada a objetos al desarrollador Web.

ASP.NET también proporciona servicios de infraestructura, como la administración de estado y el reciclaje de procesos, que reduce aún más la cantidad de código que debe escribir el desarrollador y aumenta la confiabilidad de la aplicación. Asimismo, ASP.NET utiliza estos mismos conceptos para permitir a los desarrolladores la entrega de software como un servicio. Al utilizar características de Servicios Web XML, los desarrolladores de ASP.NET pueden escribir su lógica empresarial y utilizar la infraestructura de ASP.NET para entregar ese servicio a través de SOAP.

### **Ingeniería de software**

Es una actividad que usa modelos para buscar un recurso aceptable en base a la adquisición de conocimientos para la solución de problemas.

La Ingeniería del Software abarca tres elementos claves que facilitan el proceso del desarrollo de software y permiten establecer bases para construir programas de alta calidad de una forma productiva; estos componentes son:



- ✚ Métodos
- ✚ Herramientas
- ✚ Procedimientos

**Métodos:** los métodos indican cómo construir técnicamente el Software, estos abarcan:

- ✚ Planificación y estimación de proyectos
- ✚ Análisis de requisitos del sistema y del software
- ✚ Diseño de estructuras de datos
- ✚ Diseño de programas y procedimientos algorítmicos
- ✚ Codificación
- ✚ Prueba
- ✚ Mantenimiento

**Herramientas:** Existen herramientas para cada uno de los métodos indicados anteriormente, cuando la información creada por una herramienta puede ser utilizada por otra se establece un sistema para el desarrollo del software llamado *Ingeniería del Software Asistida por Ordenador*; cada método tiene su propia herramienta.

**Procedimientos:** son las aplicaciones de los métodos utilizando las herramientas, definen la secuencia en la que se emplean los métodos, ayudan a asegurar la calidad y coordinar los cambios, etc.

## Metodología orientada a objetos

La metodología orientada a objetos se ha derivado de las metodologías anteriores a ésta, así también, los métodos de diseño estructurado guían a los desarrolladores que construyen sistemas complejos utilizando algoritmos como sus bloques fundamentales de construcción.

Similarmente los métodos de diseño orientado a objetos han evolucionado para ayudar a los programadores a explotar el poder de los lenguajes de programación basados en objetos y orientados a objetos, utilizando las clases y objetos como bloques de construcción básicos.

La Programación Orientada a Objetos, POO (Object Oriented Programming), ha probado ser un concepto uniforme en las ciencias de la computación, aplicable no sólo a los lenguajes de programación sino también al diseño de interfaces de usuario, bases de datos y arquitectura de computadoras por completo. La razón de ello es, simplemente, que una orientación a objetos nos ayuda a hacer frente a la complejidad de los diferentes sistemas.

Se define a un *objeto* como "una entidad tangible que muestra alguna conducta bien definida". Un objeto "es cualquier cosa, real o abstracta, acerca de la cual almacenamos datos y los métodos que controlan dichos datos".

Los objetos tienen una cierta "integridad" la cual no deberá ser violada. En particular, un objeto puede solamente cambiar el estado, conducta, ser manipulado o estar en relación con otros objetos de manera apropiada a este objeto.

Actualmente, el Análisis Orientado a Objetos (AOO) va progresando como método de análisis de requisitos y como complemento de otros métodos de análisis.

En lugar de examinar un problema mediante el modelo clásico de entrada-proceso-salida (flujo de información) o mediante un modelo derivado exclusivamente de estructuras jerárquicas de información, el AOO introduce varios conceptos nuevos. Esta concepción es considerada inusual para mucha gente, sin embargo su uso es de gran aplicación.

Una **clase** es una plantilla para objetos múltiples con características similares de un conjunto particular de objetos, cuando se crea un programa en lenguaje orientado a objetos, no se definen objetos verdaderos sino se definen clases de objetos.

Una **instancia** de una clase es otro término para un objeto real, si la clase es la representación general de un objeto, una instancia es su representación concreta. A menudo se utiliza indistintamente la palabra objeto o instancia para referirse, precisamente, a un objeto.

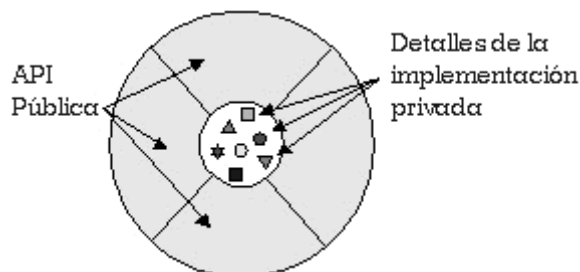
En los lenguajes orientados a objetos, cada clase está compuesta de dos cualidades: *atributos* (estado) y *métodos* (comportamiento o conducta).

Los atributos son las características individuales que diferencian a un objeto de otro (ambos de la misma clase) y determinan la apariencia, estado u otras cualidades de ese objeto. Los atributos de un objeto incluyen información sobre su estado.

Los métodos de una clase determinan el comportamiento o conducta que requiere como elemento esencial esa clase, para que sus instancias puedan cambiar su estado interno o cuando dichas instancias son llamadas para realizar un proceso por otra clase o instancia.

Los atributos se encuentran en la parte interna mientras que los métodos se encuentran en la parte externa del objeto.

Representación visual de un objeto como componente de software



Se crean métodos para definir el comportamiento de un objeto, tienen una apariencia y se comportan igual a las funciones que en otros lenguajes estructurados de programación, se definen dentro de una clase.

Los métodos no siempre afectan a un solo objeto, se comunican entre sí a través de ellos. Una clase u objeto pueden utilizar métodos en otra clase u objeto para alertar sobre los cambios en el ambiente o para solicitarle a ese objeto que cambie su estado.

Cualquier cosa que un objeto no conoce, o no puede hacer, es excluida del objeto. Además, como se puede observar en los diagramas, las variables del objeto se localizan en el centro o núcleo del objeto.

Los métodos rodean y esconden el núcleo de un objeto de otros objetos en el programa. Al empaquetamiento de las variables de un objeto con la protección de sus métodos se le llama encapsulamiento. Típicamente el encapsulamiento es utilizado para esconder detalles sin importancia de la puesta en práctica de otros objetos, pudiendo cambiar en cualquier tiempo sin afectar otras partes del programa.

Esta imagen conceptual de un objeto —un núcleo de variables empaquetadas en una membrana protectora de métodos— es una representación acertada de un objeto y es el modelo por el que luchan los diseñadores de sistemas orientados a objetos.

Sin embargo, no lo es todo, a menudo, por razones de eficiencia o la puesta en práctica, un objeto puede exponer algunas de sus variables o esconder algunos de sus métodos.

El encapsulamiento de variables y métodos en un componente de software ordenado es, todavía, una simple idea poderosa que provee dos principales beneficios a los desarrolladores de software:

**Modularidad:** el código fuente de un objeto puede ser escrito, así como darle mantenimiento, independientemente del código fuente de otros objetos. Así mismo, un objeto puede ser transferido alrededor del sistema sin alterar su estado y conducta.

**Ocultamiento de la información:** un objeto tiene una "interfaz pública" que otros objetos utilizan para comunicarse con él, además puede mantener información y métodos privados que tienen la posibilidad de ser cambiados en cualquier tiempo sin afectar a los otros objetos que dependan de ello.

Las clases proveen el beneficio de la *reutilización*, los programadores de software utilizan la misma clase, y por lo tanto el mismo código, una y otra vez para crear muchos objetos.

En las implantaciones orientadas a objetos, se percibe un objeto como un paquete de datos y procedimientos que se pueden llevar a cabo con estos datos; esto encapsula los datos y los procedimientos.

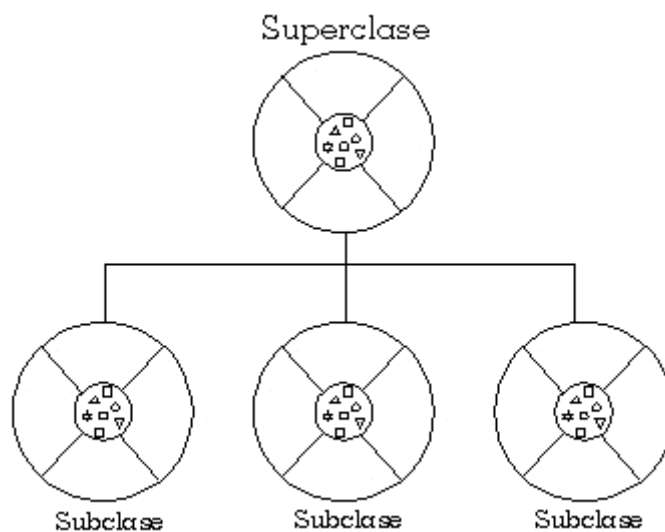
La realidad es diferente, los atributos se relacionan al objeto o instancia y los métodos a la clase, actúan de esta forma porque los atributos son variables comunes en cada objeto de una clase y cada uno de ellos puede tener un valor asociado para cada variable, diferente al que tienen para esa misma variable los demás objetos.

Los métodos, por su parte, pertenecen a la clase y no se almacenan en cada objeto, puesto que sería un desperdicio guardar el mismo procedimiento varias veces y ello va contra el principio de reutilización de código.

Otro concepto muy importante en la metodología orientada a objetos es la *herencia*. La herencia es un mecanismo poderoso con el cual se puede definir una clase en términos de otra clase; lo que significa que cuando se escribe una clase, sólo se tiene que especificar la diferencia de esa clase con otra, con lo cual, la herencia dará acceso automático a la información contenida en esa otra clase.

Con la herencia, todas las clases están arregladas dentro de una jerarquía estricta, cada clase tiene una superclase (la clase superior en la jerarquía) y puede tener una o más subclases (las clases que se encuentran debajo de esa clase en la jerarquía). Se dice que las clases inferiores en la jerarquía, las clases hijas, heredan las características de las clases más altas, las clases padres.

Las subclases heredan todos los métodos y variables de las superclases si la superclase define un comportamiento que la clase hija necesita, no se tendrá que redefinir o copiar ese código de la clase padre.



De esta manera, se puede pensar en una jerarquía de clase como la definición de conceptos demasiado abstractos en lo alto de la jerarquía y esas ideas se convierten en algo más concreto conforme se desciende por la cadena de la superclase.

Sin embargo, las clases hijas no están limitadas al estado y conducta provistos por sus superclases; pueden agregar variables y métodos además de los que ya heredan de sus clases padres. Las clases hijas pueden, también, sobrescribir los métodos que heredan por implementaciones especializadas para esos métodos. De igual manera, no hay limitación a un sólo nivel de herencia por lo que se tiene un árbol de herencia en el que se puede heredar varios niveles hacia abajo y mientras más niveles descienda una clase, más especializada será su conducta.

La herencia presenta los siguientes beneficios:

- ✚ Las subclases proveen conductas especializadas sobre la base de elementos comunes provistos por la superclase. A través del uso de herencia, los programadores pueden reutilizar el código de la superclase muchas veces.

- ✚ Los programadores pueden implementar superclases llamadas clases abstractas que definen conductas "genéricas", las superclases abstractas definen, y pueden implementar parcialmente, la conducta pero gran parte de la clase no está definida ni implementada. Otros programadores concluirán esos detalles con subclases especializadas.

### **Ventajas de la metodología orientada a objetos**

En síntesis, algunas ventajas que presenta son:

**Reutilización.** Las clases están diseñadas para que se reutilicen en muchos sistemas, maximizando la reutilización, se construyen de manera que se puedan adaptar a otros sistemas. Un objetivo fundamental de las técnicas orientadas a objetos es lograr la reutilización masiva al construir el software.

**Estabilidad.** Las clases diseñadas para una reutilización repetida se vuelven estables, de la misma manera que los microprocesadores y los chips.

El diseñador piensa en términos del comportamiento de objetos y no en detalles de bajo nivel, por su parte el encapsulamiento oculta los detalles y hace que las clases complejas sean fáciles de utilizar.

Se construyen clases cada vez más complejas, a partir de otras clases, las cuales a su vez se integran mediante éstas. Esto permite construir componentes de software que a su vez se convierten en bloques de construcción de software más complejo.



**Calidad.** Los diseños suelen tener mayor calidad, puesto que se integran a partir de componentes probados, que han sido verificados y pulidos varias veces.

**Diseño más rápido.** Las aplicaciones se crean a partir de componentes ya existentes, muchos de los componentes están contruidos de modo que se pueden adaptar a un diseño particular.

**Integridad.** Las estructuras de datos (los objetos) sólo se pueden utilizar con métodos específicos, tienen particular importancia en los sistemas cliente-servidor y los sistemas distribuidos, impidiendo que usuarios desconocidos accedan al sistema.

**Mantenimiento más sencillo.** El programador encargado del mantenimiento cambia un método de clase a la vez, cada clase efectúa sus funciones independientemente de las demás.

**Interfaz de pantalla sugestiva para el usuario.** Hay que utilizar una interfaz gráfica de modo que el usuario apunte a iconos o elementos de un menú desplegado, relacionados con los objetos; ver y apuntar es más fácil que recordar y escribir.

**Independencia del diseño.** Las clases están diseñadas para ser independientes del ambiente de plataformas, hardware y software; utilizan solicitudes y respuestas con formato estándar. Esto les permite ser requeridas en múltiples sistemas operativos, controladores de bases de datos, controladores de red, interfaces de usuario gráficas, etc. El creador del software no tiene que preocuparse por el ambiente o esperar a que éste se especifique.

**Interacción.** El software de varios proveedores puede funcionar como conjunto, un proveedor utiliza clases de otros. Existe una forma estándar de localizar clases e interactuar con ellas, el software desarrollado de manera independiente en lugares ajenos debe poder funcionar en forma conjunta y aparecer como una sola unidad ante el usuario.

**Computación Cliente-Servidor.** En los sistemas cliente-servidor, las clases en el software cliente deben enviar solicitudes a las clases en el software servidor y recibir respuestas. Una clase servidor puede ser utilizada por diferentes clientes, estos clientes sólo pueden tener acceso a los datos del servidor a través de los métodos de la clase, por lo tanto los datos están protegidos contra su corrupción.

**Computación de distribución masiva.** Las redes a nivel mundial utilizarán directorios de software de objetos accesibles, el diseño orientado a objetos es la clave para la computación de distribución masiva. Las clases de una máquina interactúan con las de algún otro lugar sin saber donde residen tales clases, ellas reciben y envían mensajes orientados a objetos en formato estándar.

**Mayor nivel de automatización de las bases de datos.** Las estructuras de datos (los objetos) en las bases de datos orientadas a objetos están ligadas a métodos que llevan a cabo acciones automáticas. Una base de datos OO tiene integrada una inteligencia, en forma de métodos, en tanto que una base de datos relacional básica carece de ello.

**Migración.** Las aplicaciones ya existentes, sean orientadas a objetos o no, pueden preservarse si se ajustan a un contenedor orientado a objetos, de modo que la comunicación con ella sea a través de mensajes estándar orientados a objetos.

**Mejores herramientas CASE.** Las herramientas CASE (Computer Aided Software Engineering, Ingeniería de Software Asistida por Computadora) utilizarán las técnicas gráficas para el diseño de las clases y de la interacción entre ellas, para el uso de los objetos existentes adaptados a nuevas aplicaciones.

Las herramientas deben facilitar el modelado en términos de eventos, formas de activación, estados de objetos, etc.

Las herramientas OO del CASE deben generar un código tan pronto se definan las clases y permitir al diseñador utilizar y probar los métodos recién creados. Las herramientas se deben diseñar de manera que apoyen el máximo de creatividad y una continua afinación del diseño durante la construcción.

### ***RUP Proceso de ingeniería del software.***

Proporciona un acercamiento disciplinado a la asignación de tareas y responsabilidades en una organización de desarrollo, su propósito es asegurar la producción de software de alta calidad que se ajuste a las necesidades de sus usuarios finales con unos costos y calendario predecibles.

En definitiva el RUP es una metodología de desarrollo de software que intenta integrar todos los aspectos a tener en cuenta durante el ciclo de vida del software, con el objetivo de hacer factibles tanto pequeños como grandes proyectos software.

Las características principales de RUP son:

- ✚ *Guiado/Manejado por casos de uso:* La razón de ser de un sistema software es servir a usuarios ya sean humanos u otros sistemas; un caso de uso es una interacción que el software provee a sus usuarios.
- ✚ *Centrado en arquitectura:* Es como una radiografía del sistema que estamos desarrollando, lo suficientemente completa para que todos los implicados en el desarrollo tengan una idea clara de qué es lo que están construyendo.
- ✚ *Iterativo e incremental:* Al proyecto se lo divide en mini-proyectos y se van creando versiones de los mismos, está centrado en la arquitectura y guiado por los casos de uso. Incluye artefactos (que son los productos tangibles del proceso como ser un modelo de casos de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso).

El RUP divide el proceso de desarrollo en ciclos, teniendo un producto al final de cada ciclo, a su vez, cada ciclo se divide en fases que finalizan con un hito donde se debe tomar una decisión importante:

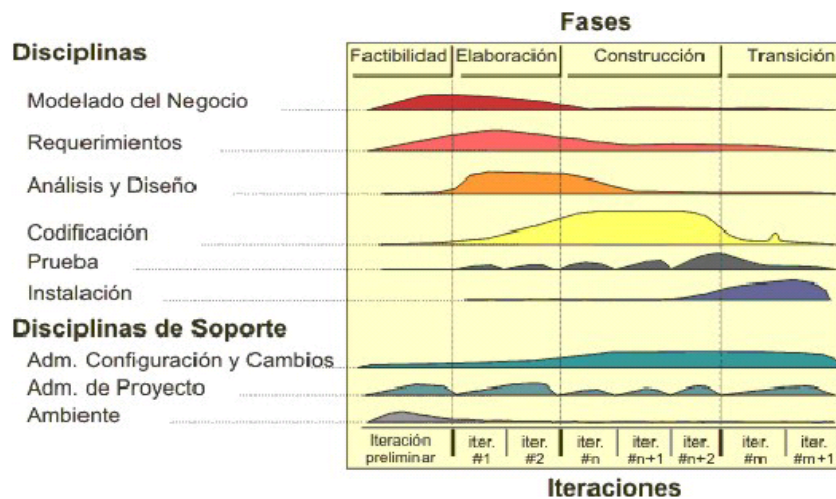
***Inicio:*** elaboración de un plan de fases, identificación de los principales casos de uso y los riesgos.

***Elaboración:*** establecer un plan de proyecto, completar los casos de uso y eliminar los riesgos.

***Construcción:*** se concentra en la elaboración de un producto eficiente, se obtiene el producto con sus aplicaciones funcionando y el manual de usuario.

**Transición:** se implementa el producto en el cliente, se hace conocer a los usuarios y como consecuencia de esto surgen nuevos requerimientos a ser analizados.

**Gráfico 4**  
**Fases del RUP**



**Fuente:** Wikimedia

## UML Lenguaje de Modelo Unificado

Es un lenguaje de modelado visual que sirve para especificar, visualizar, construir y documentar una de las partes que comprende el desarrollo de software. Se usa para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.

UML entrega una forma de modelar conceptualizaciones de los procesos de negocio y funciones de sistema, escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software re-usables.

UML capta la información sobre la estructura estática y el comportamiento dinámico de un sistema, es un lenguaje de propósito general para el modelado orientado a objetos

### **SQL Server Express**

SQL Server Express ayuda a los programadores a crear aplicaciones sólidas y fiables al ofrecer un sistema de base de datos robusto, gratuito y fácil de usar.

Con frecuencia los sistemas de bases de datos son excesivamente complejos para crear aplicaciones sencillas, Visual Studio® 2005 y SQL Server Express reducen esta complejidad proporcionando un entorno sencillo pero eficaz para crear aplicaciones basadas en datos.

Los programadores pueden diseñar esquemas, agregar datos y realizar consultas en bases de datos locales, todo ello desde el entorno de Visual Studio 2005, Si necesitan características de bases de datos más avanzadas, SQL Server Express puede actualizarse sin problemas a versiones más sofisticadas.

SQL Server Express es una herramienta nueva y ligera de fácil instalación, configuración, uso, administración y edición de consultas, proporciona una rápida descarga y optimiza el tiempo de consultas, su implementación integrada se realiza mediante un solo Clic.

Esta base de datos es compatible con: Active Directory, la autenticación de Windows, T-SQL, datos estructurados y semiestructurados, XQuery, esquemas XML.

SQL Server Express es gratuito porque fue creado en base a la filosofía de ayudar a los programadores a crear excelentes aplicaciones contribuyendo a conseguir una fidelidad de los clientes a largo plazo.

Proporciona un administrador de equipo para iniciar y detener el servicio, derechos de administración detallados, tres niveles de seguridad de acceso al código, acceso externo (verificable), aprovechamiento de los procedimientos almacenados como capa de abstracción, redistribución libre de cánones (requiere registro), amplia funcionalidad de bases de datos, servicio y aplicación de revisiones automatizados (parte de Microsoft Update), procedimientos almacenados, vistas, triggers, cursores, índices ampliados y aislamiento a nivel de snapshot.

Está integrado con Visual Studio 2005, utiliza el explorador de datos de Visual Studio para diseñar esquemas, agregar datos y realizar consultas en bases de datos locales, maneja tipos y agregados definidos por el usuario, depuración integrada, controles de informes de SQL en VS; es suscriptor de réplica transaccional y de réplica de combinación y cliente para SQL Server Service bróker.

### **2.3. MARCO ESPACIAL**

La realización del proyecto se ejecutó en un lapso de 12 meses, a partir de la aprobación, colocando especial énfasis en la adaptabilidad del sistema informático a las reformas a Ley de Régimen Tributario y con la posibilidad de acoger reformas futuras.

Este sistema cuenta con un amplio componente legal desde el punto de vista tributario, asegurando la fiabilidad del cálculo del impuesto con estricto cumplimiento de las exigencias legales.

Al ser un software orientado a la Web es de fácil acceso para el beneficiario del mismo. Permite determinar el Impuesto a la renta de una persona natural, que realice actividades empresariales y que operen con un capital propio inferior a USD 60.000, o sus ingresos brutos anuales sean inferiores a \$100.000, o sus costos y gastos no superen los USD 80.000, así como los profesionales, agentes o comisionistas, trabajadores autónomos y artesanos.

## **2.4. MARCO LEGAL**

### **2.4.1 Disposiciones Legales**

En la Constitución actual del Ecuador resaltan los siguientes artículos que hacen referencia al desarrollo y distribución del software.

#### **Derechos, Sección tercera**

**Art. 16.** Todas las personas, en forma individual o colectiva, tienen derecho a:

2. El acceso universal a las tecnologías de información y comunicación.



## **Capítulo VI. Trabajo y producción, Sección II. Tipos de propiedad**

**Art. 322.** Se reconoce la propiedad intelectual de acuerdo con las condiciones que señala la ley. Se prohíbe toda forma de apropiación de conocimientos colectivos, en el ámbito de las ciencias, tecnologías y saberes ancestrales...

La propiedad intelectual tiene que ver con las creaciones de la mente, se divide en dos categorías: la propiedad industrial, que incluye las invenciones, patentes, marcas, dibujos y modelos industriales e indicaciones geográficas de origen; y el derecho de autor, que abarca las obras literarias y artísticas, tales como las novelas, los poemas y las obras de teatro, las películas, las obras musicales, las obras de arte, tales como los dibujos, pinturas, fotografías y esculturas, y los diseños arquitectónicos.

## **Sección VIII. Ciencia, tecnología, innovación y saberes ancestrales.**

**Art. 385.** El sistema nacional de ciencia, tecnología, innovación y saberes ancestrales, en el marco del respeto al ambiente, la naturaleza, la vida, las culturas y la soberanía, tendrá como finalidad:

Desarrollar tecnologías e innovaciones que impulsen la producción nacional, eleven la eficiencia y productividad, mejoren la calidad de vida y contribuyan a la realización del buen vivir.

**La Ley de Propiedad Intelectual Ecuatoriana, en su Art. 7.**, define como “Programa de ordenador (software): Toda secuencia de instrucciones o indicaciones destinadas a ser utilizadas, directa o indirectamente, en un dispositivo de lectura automatizada, ordenador, o aparato electrónico o similar con capacidad de procesar información, para la realización de una función o tarea, u obtención de un resultado determinado, cualquiera que fuere su forma de expresión o fijación. El programa de ordenador comprende también la documentación preparatoria, planes y diseños, la documentación técnica y, los manuales de uso.”

El **Art. 28** del mismo cuerpo legal, expresa que “los programas de ordenador se consideran obras literarias y se protegen como tales... independientemente de que hayan sido incorporados en un ordenador y cualquiera sea la forma en que estén expresados...”

Para poder establecer concretamente los efectos causados por la reproducción de una obra para su uso personal, es necesario que previamente aclaremos que es “copia privada” según nuestra Ley “la copia doméstica de fonogramas o video gramas, o la reproducción reprográfica en un solo ejemplar realizada por el adquirente original de un fonograma o video grama u obra literaria de circulación lícita, destinada exclusivamente para el uso no lucrativo de la persona natural que la realiza...”

#### **2.4.2. El software como objeto protegido**

Los objetos de protección que presenta el software son los siguientes:

***Documentación preparatoria del software:*** Materiales de concepción, los diagramas de flujo, las especificaciones formales; además, se le asimilan al rol que en las ciencias desempeñan las topografías, gráficos, mapas y diseños.

***Programa computacional o también llamado conjunto de instrucciones:*** Expresión de un conjunto de instrucciones mediante palabras, códigos, planes o en cualquier otra forma que, al ser incorporadas en un dispositivo de lectura automatizada, es capaz de hacer que un computador ejecute una tarea u obtenga un resultado.

***Documentación técnica y manuales de uso (documentación auxiliar):***

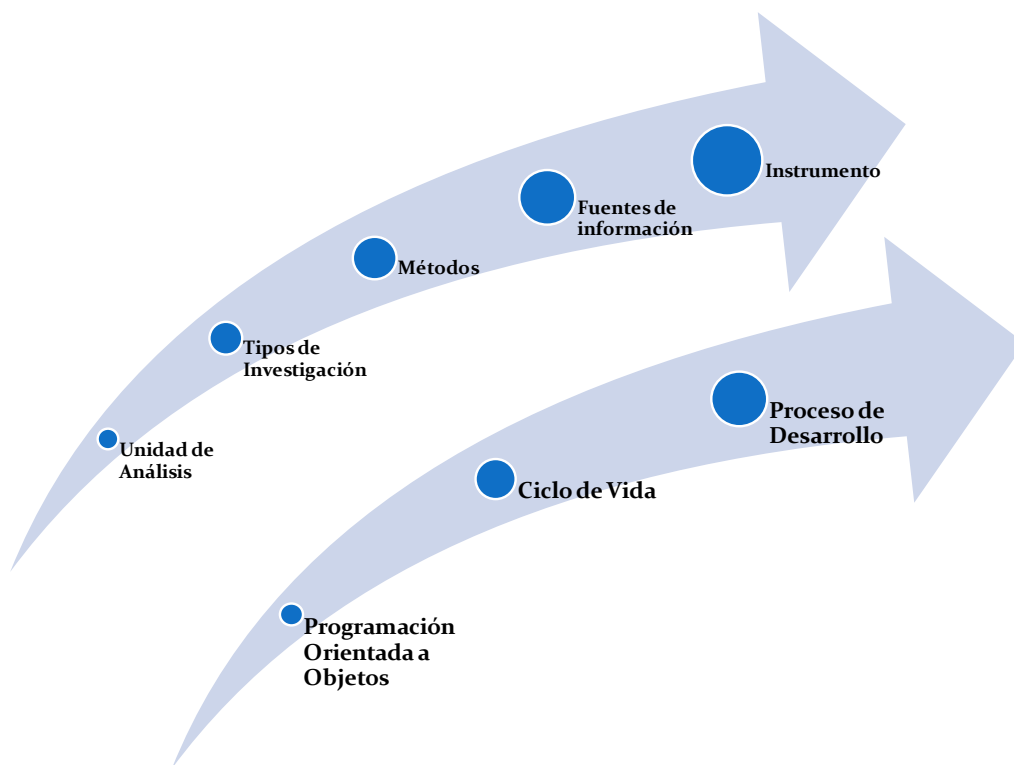
Documentación técnica: Información sobre condiciones técnicas de instalación y utilización, sobre corrección de errores; sobre su puesta al día, etc.

Manuales de uso: literatura destinada al usuario para facilitarle la operación del programa.

El desarrollo de software en el Ecuador es una industria de nueva creación por lo cual cuenta con una escueta regulación dentro de la legislación ecuatoriana.

## 3. METODOLOGIA

### 3.1. METODOLOGÍA DE INVESTIGACIÓN



#### 3.1.1. Unidad de Análisis

La unidad de análisis son las personas naturales que realicen actividades empresariales, así como los profesionales, comisionistas, artesanos, agentes, representantes y demás trabajadores autónomos que no estén obligados a llevar contabilidad, que deban realizar sus declaraciones de Impuesto a la Renta.

Los usuarios podrán acceder con facilidad a este sistema con el fin de realizar su registro de ingresos y egresos para determinar su impuesto de acuerdo a los preceptos legales establecidos para el efecto.

### 3.1.2. Tipos de Investigación

La metodología utilizada incluye los siguientes tipos de investigación:

***Investigación de campo:*** la investigación para el desarrollo de la aplicación web se apoya en la recopilación de información mediante entrevistas, cuestionarios, encuestas y observaciones.

***Investigación aplicada o tecnológica:*** Está dirigida a la aplicación de las teorías y las leyes, con la finalidad de resolver problemas sociales.

***Investigación bibliográfica:*** Recopilación de información al inicio y durante el desarrollo del proyecto, Se acudió a la revisión bibliográfica de textos, archivos de información y citas de internet, páginas web, etc.

### 3.1.3. Métodos

Para el desarrollo de este proyecto se utilizó el método científico, cumpliendo con las siguientes fases:

- Elección del problema
- Planteamiento del problema
- Revisión bibliográfica
- Elección de técnicas
- Recolección de datos
- Análisis de datos
- Conclusiones y recomendaciones

### **3.1.4 Fuentes de información**

Las fuentes primarias y/o secundarias para la recolección de información de este proyecto, se ubicaron en la bibliografía, además se realizaron entrevistas a 50 personas sujetos pasivos del Impuesto a la Renta no obligadas a llevar contabilidad.

### **3.1.5. Técnicas**

Se empleó la Entrevista a un número o muestra representativa de la población que utiliza o necesita utilizar una herramienta informática para declarar sus impuestos, con el fin de conocer sus opiniones, aspiraciones o inquietudes.

Con esta técnica se obtuvo información mediante un conjunto de preguntas formuladas a través de la interrelación verbal entre el entrevistador y los entrevistados.

### **3.1.6. Instrumento**

Se utilizaron Fichas de Trabajo de contenido y fichas textuales para registrar los datos e información encontrada al consultar las fuentes de información (lectura de libros, revistas, periódicos, consulta a personas, documentos de trabajo).

Ejemplo:

Universidad Tecnológica Israel

**Sistema de Ingresos y Egresos para el Cálculo del Impuesto a la Renta para Personas  
Naturales No Obligadas a Llevar Contabilidad**

**FICHA DE CONTENIDO**

Lugar:	No.
Fuente:	Investigador:
Título:	Fecha:
Contenido	

Universidad Tecnológica Israel

**Sistema de Ingresos y Egresos para el Cálculo del Impuesto a la Renta para Personas**

**Naturales No Obligadas a Llevar Contabilidad**

**ENTREVISTA A PERSONAS NATURALES NO OBLIGADAS**

**A LLEVAR CONTABILIDAD**

**Nombre:** \_\_\_\_\_

**Edad:** \_\_\_\_\_

**Profesión:** \_\_\_\_\_

1.- ¿Sabe usted si existe un sistema informático para declarar el Impuesto a la Renta?

SI  NO

2.- ¿Sabe usted como declarar el Impuesto a la Renta?

SI  NO

3.- ¿Sabe usted cuando tiene que declarar el Impuesto a la Renta?

SI  NO

4.- ¿Sabe usted cuánto tiene que pagar por Impuesto a la Renta?

SI  NO



5.- ¿Sabe usted cuando se generan multas e intereses?

SI  NO

6.- ¿Sabe usted cuando se generan pagos indebidos o en exceso?

SI  NO

7.- ¿Sabe usted cuáles son los requisitos que exige el SRI para declarar el Impuesto a la Renta?

SI  NO

8.- ¿Cree que es necesario automatizar el proceso para declarar el Impuestos a la Renta?

SI  NO

9.- ¿Cree que es una buena opción crear una aplicación Web a la que se puede acceder desde cualquier lugar?

SI  NO

## **3.2. METODOLOGÍA INFORMÁTICA**

### **3.2.1. Metodología**

La metodología utilizada es la “Metodología **Orientada a Objetos**”, ya que permite incluir varias técnicas, como herencia, modularidad, polimorfismo, y encapsulamiento. Además que en la actualidad son muchos los lenguajes de programación que soportan la orientación a objetos.

La metodología orientada a objetos expresa un programa como un conjunto de estos objetos, que colaboran entre ellos para realizar tareas, esto permite hacer los programas y módulos más fáciles de escribir, mantener y reutilizar.

De esta forma, un objeto contiene toda la información que permite definirlo e identificarlo frente a otros objetos pertenecientes a otras clases e incluso frente a objetos de una misma clase, al tener valores bien diferenciados en sus atributos. A su vez, los objetos disponen de mecanismos de interacción (métodos) que favorecen la comunicación entre ellos.

Esta comunicación favorece a su vez el cambio de estado en los propios objetos. Esta característica lleva a tratarlos como unidades indivisibles, en las que no se separan ni deben separarse el estado y el comportamiento.

### **3.2.2. Ciclo de Vida**

Proceso unificado de desarrollo (RUP).

**Cuadro 18**  
**Ciclo de vida del proyecto**

<b>Descripción</b>	<b>Hito</b>
<b>Fase de Inicio</b>	En esta fase se desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el software. Los principales casos de uso serán identificados.
<b>Fase de Elaboración</b>	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera liberación de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase.
<b>Fase de Construcción</b>	Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a una iteración, produciendo una liberación a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la primera versión, lista para ser entregada a los usuarios para pruebas beta.
<b>Fase de Transición</b>	En esta fase se preparará la versión para distribución, asegurando una implantación, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y todo el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.

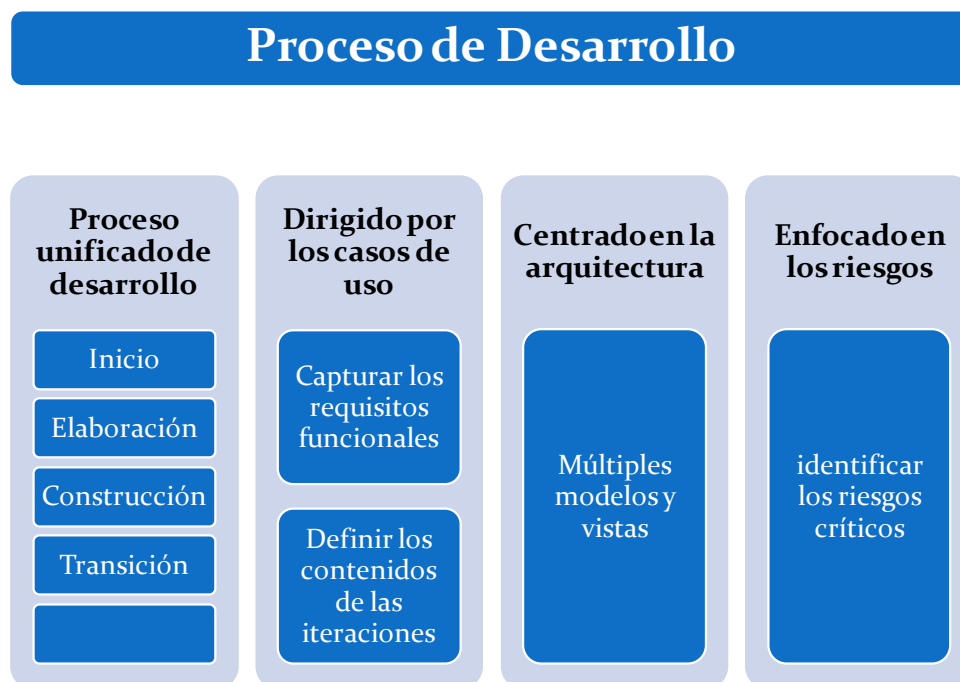
**Fuente:** Rafael Chumbi

La **documentación** es una tarea importante que se realiza en todas las etapas, cada etapa tiene como entrada uno o varios documentos procedentes de las etapas anteriores y produce otros documentos de salida.

### 3.2.3. Proceso de Desarrollo

#### 3.2.3.1. Proceso unificado de desarrollo

El Proceso que se seguirá es el Proceso Unificado puesto que es un marco de desarrollo iterativo e incremental compuesto de cuatro fases denominadas Inicio, Elaboración, Construcción y Transición. Cada una de estas fases estará dividida en una serie de iteraciones que darán como resultado un *incremento* del producto, estas iteraciones se dividen en Análisis de requisitos, Diseño, Implementación y Prueba.



#### 3.2.3.2. Dirigido por los casos de uso

En el Proceso Unificado los casos de uso se utilizan para capturar los requisitos funcionales y definir los contenidos de las iteraciones que menciona el tema.

### **3.2.3.3. Centrado en la arquitectura**

El Proceso Unificado asume que no existe un modelo único que cubra todos los aspectos del sistema, por tal motivo existen múltiples modelos y vistas que definen la arquitectura de software de un sistema.

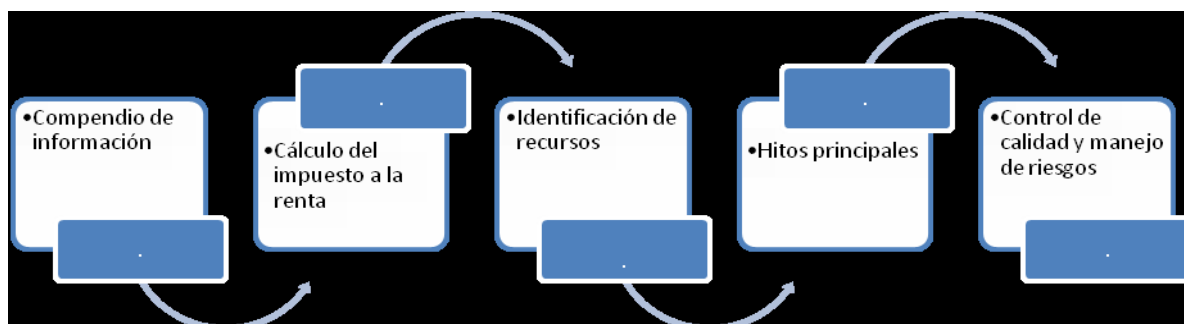
### **3.2.3.4. Enfocado en los riesgos**

El Proceso Unificado requiere que el equipo del proyecto se centre en identificar los riesgos críticos en una etapa temprana del ciclo de vida. Los resultados de cada iteración, en especial los de la fase de Elaboración, deben ser seleccionados en un orden que asegure que los riesgos principales son considerados primero.

## 4. PROCESO DE DESARROLLO

### 4.1 PLAN DE DESARROLLO DE SOFTWARE

El Plan de Desarrollo de Software es un compendio de toda la información requerida para la Gestión del Proyecto de software informático para el cálculo del impuesto a la renta de personas naturales no obligadas a llevar contabilidad. Identifica recursos, trabajadores, clientes y usuarios de referencia; indicando su información de contacto y participación en el proyecto; sin olvidar los hitos principales y las consideraciones sobre el control de calidad y manejo de riesgos, entre otras, que son pertinentes en el desarrollo.



#### 4.1.1. Propósito

El propósito del Plan de Desarrollo de Software es proporcionar la información necesaria para controlar el proyecto, además nos permite identificar recursos, trabajadores, clientes y usuarios, en él se describe el enfoque de desarrollo del software.

#### 4.1.2 Alcance

Para la versión 1.0 del Plan de Desarrollo del Software, se ha recolectado los requisitos por medio de los instrumentos de investigación como son: fichas de observación y encuestas, para hacer una estimación aproximada, una vez comenzado el proyecto y durante la fase de Inicio se generará la primera versión del artefacto “Visión”. La utilización de este Plan de Proyecto sirve para obtener una visión fresca, actual y completa de la organización del esfuerzo que hay por detrás del proyecto, hay que tener presente que luego de cada hito importante este documento genera una nueva versión.

#### 4.1.3 Resumen

Un poco de contexto ayuda a entender las cosas mejor, por lo que es prudente hacer un resumen de las metas y fundamentos del proyecto quedando conformado de la siguiente manera:

**Vista General del Proyecto.-** proporciona una descripción del propósito, alcance y objetivos del proyecto, estableciendo los artefactos que serán producidos y utilizados durante el proyecto.

**Organización del Proyecto.-** describe la estructura organizacional del equipo de desarrollo.

**Gestión del Proceso.-** explica los costos y planificación estimada, define las fases e hitos del proyecto y describe cómo se realizará su seguimiento.

**Planes y Guías de aplicación.-** proporciona una vista global del proceso de desarrollo de software, incluyendo métodos, herramientas y técnicas que serán utilizadas.

#### **4.1.4 Vista general del proyecto**

##### **4.1.4.1 Propósito, alcance y objetivos**

El **Sistema de Ingresos y Egresos para el Cálculo del Impuesto a la Renta para Personas Naturales No Obligadas a Llevar Contabilidad**, responde a una necesidad latente en los contribuyentes sujetos al impuesto a la renta en el Ecuador, asegurando su aplicación y pertinencia.

En el campo tributario no se han desarrollado en mayor magnitud herramientas informáticas por los cambios abruptos en la aplicación de las normas establecidas por la administración tributaria, siendo interesante la especialización en este campo.

Es de suma importancia investigar este tema ya que contaría con un gran universo de usuarios, puesto que la cultura tributaria está tomando fuerza en el país por las disposiciones legales existentes, además que el incumplimiento de la normatividad vigente podría traer consecuencias inclusive de orden penal.

Es necesario contar con una herramienta flexible y de rápida adaptabilidad a los cambios para que pueda ágilmente incorporar los cambios que se establezcan en la ley y que permita su fácil aplicación.



#### 4.1.4.2 Suposiciones y restricciones

Las suposiciones y restricciones que el sistema puede tener y que se obtuvieron a partir de encuestas a los usuarios son las siguientes:

Debe contemplarse las implicaciones de los siguientes puntos críticos:

- ✚ Generación de reportes de ingresos y egresos.
- ✚ Uso compartido de la aplicación.
- ✚ Sistemas seguros: protección de información, seguridad en las transmisiones de datos (PKI), etc.
- ✚ Gestión de flujos de trabajo, seguridad de transacciones e intercambio de información
- ✚ Adaptación a la normativa de Protección de Datos
- ✚ La automatización del cálculo del impuesto debe ajustarse a la legislación vigente y considerar la previsión de la nueva legislación.

Como es natural, la lista de suposiciones y restricciones se incrementará durante el desarrollo del proyecto, particularmente una vez establecido el artefacto “Visión”.

#### 4.1.4.3 Entregables del proyecto

A continuación se indican y describen cada uno de los artefactos que serán generados y utilizados por el proyecto y que constituyen los entregables. Esta lista constituye la configuración de RUP desde la perspectiva de artefactos, y que se propone para este proyecto.

Es preciso destacar que de acuerdo a la filosofía de RUP (y de todo proceso iterativo e incremental), todos los artefactos son objeto de modificaciones a lo largo del proceso de desarrollo, con lo cual, sólo al término del proceso podríamos tener una versión definitiva y completa de cada uno de ellos. Sin embargo, el resultado de cada iteración y los hitos del proyecto están enfocados a conseguir un cierto grado de completitud y estabilidad de los artefactos. Esto será indicado más adelante cuando se presenten los objetivos de cada iteración.

### **Plan de Desarrollo del Software**

El plan de Desarrollo del proyecto, se lo presentará a lo largo de lo enunciado en el presente documento.

### **Modelo de Casos de Uso del Negocio**

Es un modelo de las funciones de negocio vistas desde la perspectiva de los actores externos (Agentes de registro, solicitantes finales, otros sistemas etc.). Permite situar al sistema en el contexto organizacional haciendo énfasis en los objetivos en este ámbito. Este modelo se representa con un Diagrama de Casos de Uso usando estereotipos específicos para este modelo.

### **Modelo de Objetos del Negocio**

Es un modelo que describe la realización de cada caso de uso del negocio, estableciendo los actores internos, la información que manipulan en términos generales y los flujos de trabajo (workflows) asociados.

Se utilizan Diagramas de Colaboración para representar actores externos, internos y las entidades (información) que manipulan; los Diagramas de Clases muestran gráficamente las entidades del sistema y sus relaciones; y, los Diagramas de Actividad para mostrar los flujos de trabajo.

### **Glosario**

En este documento se definen los principales términos usados en el proyecto, permitiendo utilizar un lenguaje común y una terminología consensuada.

### **Modelo de Casos de Uso**

El modelo de Casos de Uso presenta mediante Diagramas de Casos de Uso las funciones del sistema y los actores que hacen uso de ellas.

### **Visión**

Este documento define la mirada a futuro del producto desde la perspectiva del cliente, especificando sus necesidades y características, constituyendo una base en cuanto a los requisitos del sistema.

### **Especificaciones de Casos de Uso**

En los casos de uso que lo requieran, cuya funcionalidad no sea evidente o que no baste con una simple descripción narrativa, se realiza una descripción detallada utilizando una plantilla de documento, donde se incluyen: precondiciones, post-condiciones, flujo de eventos, requisitos no-funcionales asociados.

También, para casos de uso cuyo flujo de eventos sea complejo podrá adjuntarse una representación gráfica mediante un Diagrama de Actividad.

### **Especificaciones Adicionales**

Este documento capturará todos los requisitos que no han sido incluidos como parte de los casos de uso y se refieren a requisitos no-funcionales globales. Dichos requisitos incluyen: requisitos legales o normas, aplicación de estándares, requisitos de calidad del producto, tales como: confiabilidad, desempeño, etc., u otros requisitos de ambiente, tales como: sistema operativo, requisitos de compatibilidad, etc.

### **Prototipos de Interfaces de Usuario**

Se trata de prototipos que permiten al usuario tener una idea más o menos precisa de las interfaces que proveerá el sistema y así, conseguir retroalimentación respecto a los requisitos del sistema.

Estos prototipos se presentan en forma de: dibujos a mano en papel, dibujos con alguna herramienta gráfica o prototipos ejecutables interactivos, siguiendo ese orden de acuerdo al avance del proyecto. Sólo los de este último tipo serán entregados al final de la fase de Elaboración, los otros serán desechados.

Asimismo, este artefacto, será desechado en la fase de Construcción en la medida que el resultado de las iteraciones vayan desarrollando el producto final.

### **Modelo de Análisis y Diseño**

Este modelo establece la elaboración de los casos de uso en clases, pasando desde una representación en términos de análisis sin incluir aspectos de implementación hacia una representación de diseño incluyendo una orientación hacia el entorno de implementación, de acuerdo al avance del proyecto.

### **Modelo de Datos**

Prevé que la persistencia de la información del sistema está soportada por una base de datos relacional, describe la representación lógica de los datos persistentes de acuerdo con el enfoque para modelado relacional de datos.

La expresión de este modelo utiliza un Diagrama de Clases con Modelado de Datos UML, para conseguir la representación de tablas, claves, etc.

### **Modelo de Implementación**

Es una colección de componentes y subsistemas que los contienen; estos componentes incluyen: ficheros ejecutables, ficheros de código fuente, y todo otro tipo de ficheros necesarios para la implantación y despliegue del sistema.

Al final de la fase de Elaboración, este modelo es sólo una versión preliminar, posteriormente tiene un alto refinamiento.

### **Modelo de Despliegue**

Muestra el despliegue de la configuración de tipos de nodos del sistema, en los cuales se hará el despliegue de los componentes.

### **Casos de Prueba**

Cada prueba es especificada mediante un documento que establece las condiciones de ejecución, las entradas de la prueba y los resultados esperados.

Estos casos son aplicados como pruebas de regresión en cada iteración, cada caso de prueba llevará asociado un procedimiento de prueba con las instrucciones para realizarla, y dependiendo del tipo de prueba dicho procedimiento podrá ser automatizable mediante un script de prueba.

### **Solicitud de Cambio**

Los cambios propuestos para los artefactos se formalizan mediante este documento, incluyendo el seguimiento de los defectos detectados, la solicitud de mejoras o cambios en los requisitos del producto. Así se provee un registro de decisiones de cambios, de su evaluación e impacto, y se asegura que éstos sean conocidos por el equipo de desarrollo. Los cambios se establecen respecto de la última baseline establecida (el estado del conjunto de los artefactos en un momento determinado del proyecto). En nuestro caso al final de cada iteración se establecerá una baseline.

### **Plan de Iteración**

Es un conjunto de actividades y tareas ordenadas temporalmente con recursos asignados y las dependencias entre ellas; se realiza para cada iteración, y para todas las fases.

### **Evaluación de Iteración**

Este documento incluye la evaluación de los resultados de cada iteración, el grado en el cual se han conseguido los objetivos de la iteración, las lecciones aprendidas y los cambios a ser realizados.

### **Lista de Riesgos**

Este documento incluye una lista de los riesgos vigentes en el proyecto, en orden decreciente en importancia y con acciones específicas de contingencia para su mitigación.

### **Manual de Instalación**

Este documento incluye las instrucciones para realizar la instalación del producto.

### **Material de Apoyo al Usuario Final**

Corresponde a un conjunto de documentos y facilidades de uso del sistema, incluyendo: Guías del Usuario, Guías de Operación, Guías de Mantenimiento y Sistema de Ayuda en Línea

### **Producto**




Los ficheros del producto empaquetados y almacenados en un CD con los mecanismos apropiados para facilitar su instalación. El producto, a partir de la primera iteración de la fase de Construcción es desarrollado incremental e iterativamente, obteniéndose una nueva versión al final de cada iteración.

#### 4.1.2.5 Evolución del Plan de Desarrollo del Software

El Plan de Desarrollo del Software se revisará semanalmente y se volverá a reajustar al comienzo de cada iteración.

#### 4.1.5 Organización del proyecto

##### 4.1.5.1 Participantes en el proyecto

 <b>Jefe de Proyecto</b>	Rafael Chumbi
 <b>Analista de Sistemas</b>	Rafael Chumbi
 <b>Analista – Programador</b>	Rafael Chumbi
 <b>Ingeniero de Software</b>	Ing. Leopoldo Pauta

##### 4.1.5.2 Roles y responsabilidades

A continuación se describen las principales responsabilidades de cada uno de los participantes en el equipo de desarrollo durante las fases de Inicio y Elaboración, de acuerdo con los roles que desempeñan en RUP.



**Cuadro 19**  
**Roles y responsabilidades**

<b>Puesto</b>	<b>Responsabilidad</b>
Jefe de Proyecto	<p>Asigna los recursos, gestiona las prioridades, coordina las interacciones con los clientes y usuarios, y mantiene al equipo del proyecto enfocado en los objetivos.</p> <p>También establece un conjunto de prácticas que aseguran la integridad y calidad de los artefactos del proyecto. Además, se encarga de supervisar el establecimiento de la Arquitectura del sistema, Gestión de riesgos, Planificación y Control del proyecto.</p>
Analista de Sistemas	<p>Captura, especificación y validación de requisitos, interactuando con el cliente y los usuarios mediante encuestas.</p> <p>Elabora el Modelo de Análisis y Diseño, colabora en la elaboración de las pruebas funcionales y el modelo de datos.</p>
Programador	<p>Construye los prototipos, colabora en la elaboración de las pruebas funcionales, modelo de datos y en las validaciones con el usuario.</p>
Ingeniero de Software	<p>Gestiona requisitos, configuración y cambios; elabora el modelo de datos, prepara las pruebas funcionales, elabora la documentación, elabora los modelos de implementación y despliegue.</p>

**Fuente:** Rafael Chumbi

#### 4.1.6 Gestión del proceso

##### 4.1.6.1 Estimaciones del proyecto

Los recursos necesarios para el desarrollo del proyecto son los siguientes:

**Cuadro 20**  
**Costo personal**

NOMBRE DE LOS INVESTIGADORES	FORMACIÓN	FUNCIÓN EN EL PROYECTO	TIEMPO DEDICA (H/SEMANA)	DURACIÓN meses	VALOR (\$/Mes)	TOTAL (\$)
Rafael Chumbi	Tecnólogo En Sistemas	Desarrollador	20	8	100	800
Gina Cuadrado	Magister Especialista en Tributación	Asesor Tributario	5	5	50	250

**Fuente:** Rafael Chumbi

**Cuadro 21**  
**Descripción de los equipos requeridos**

EQUIPO	JUSTIFICACIÓN USO EN EL PROYECTO	NRO. UNIDAD	ADQUISICIÓN	VALOR (\$/Mes)	TOTAL (\$)
Computadora	Codificación	1	900	250	1000
Impresora	Impresiones	1	300	300	300

**Fuente:** Rafael Chumbi

**Cuadro 22**  
**Descripción de viajes**

<b>NRO</b>	<b>LUGAR</b>	<b>VALOR PASAJE</b>	<b>ESTADIA (3 DIAS)</b>	<b>JUSTIFICACION</b>	<b>VALOR (\$/Viaje)</b>	<b>TOTAL (\$)</b>
3	Quito	30	100	Rafael Chumbi	130	390

**Fuente:** Rafael Chumbi

**Cuadro 23**  
**Descripción de Insumos y Materiales**

<b>NOMBRE</b>	<b>USO</b>	<b>CANTIDAD</b>	<b>VALOR UNITARIO \$</b>	<b>TOTAL \$</b>
Resmas de papel	Imprimir trabajo final	2	5	10
Tinta	Recarga de cartuchos	1	30	30

**Fuente:** Rafael Chumbi

**Cuadro 24**  
**Descripción Servicios Técnicos**

<b>NOMBRE</b>	<b>USO</b>	<b>CANTIDAD</b>	<b>VALOR UNITARIO \$/mes</b>	<b>TOTAL \$</b>
Internet	Investigación	8	20	160
Mantenimiento	Mantenimiento equipos	1	30	30

**Fuente:** Rafael Chumbi

**Cuadro 25**  
**Costo total del proyecto**

<b>CONCEPTO</b>	<b>VALOR</b>
Costo personal	1050
Equipos	1300
Viajes	390
Insumos y materiales	40
Servicios técnicos	190
Total	2970

**Fuente:** Rafael Chumbi

#### **4.1.6.2 Plan del proyecto**

##### ***Plan de las Fases***

Este proyecto se llevará a cabo en fases con una o más iteraciones en cada una de ellas. La siguiente tabla muestra la distribución de tiempos y el número de iteraciones.

**Cuadro 26**  
**Plan de fases**

<b>Fase</b>	<b>Nro. Iteraciones</b>	<b>Duración</b>
Fase de Inicio	1	1 semana
Fase de Elaboración	2	4 semanas
Fase de Construcción	2	4 semanas
Fase de Transición	1	1 semana

**Fuente:** Rafael Chumbi

Al final de cada fase se deben tener los hitos que se describen en el siguiente cuadro.

**Cuadro 27**  
**Hitos de las fases**

<b>Descripción</b>	<b>Hito</b>
Fase de Inicio	En esta fase se desarrollarán los requisitos del producto desde la perspectiva del usuario, los cuales serán establecidos en el artefacto Visión. Los principales casos de uso serán identificados y se refinará el Plan de Desarrollo del Proyecto. La aceptación del cliente /usuario del artefacto Visión y el Plan de Desarrollo marcan el final de esta fase.
Fase de Elaboración	En esta fase se analizan los requisitos y se desarrolla un prototipo de arquitectura (incluyendo las partes más relevantes y / o críticas del sistema). Al final de esta fase, todos los casos de uso correspondientes a requisitos que serán implementados en la primera etapa de la fase de Construcción deben estar analizados y diseñados (en el Modelo de Análisis / Diseño). La revisión y aceptación del prototipo de la arquitectura del sistema marca el final de esta fase.

Fase de Construcción	Durante la fase de construcción se terminan de analizar y diseñar todos los casos de uso, refinando el Modelo de Análisis / Diseño. El producto se construye en base a 2 iteraciones, cada una produciendo una versión a la cual se le aplican las pruebas y se valida con el cliente / usuario. Se comienza la elaboración de material de apoyo al usuario. El hito que marca el fin de esta fase es la versión con la capacidad operacional parcial del producto que se haya considerado como crítica, lista para ser entregada a los usuarios para pruebas beta.
Fase de Transición	En esta fase se prepararán dos versiones para distribución, asegurando una implantación, incluyendo el entrenamiento de los usuarios. El hito que marca el fin de esta fase incluye, la entrega de toda la documentación del proyecto con los manuales de instalación y el material de apoyo al usuario, la finalización del entrenamiento de los usuarios y el empaquetamiento del producto.

**Fuente:** Rafael Chumbi

### **Calendario del Proyecto**

A continuación se presenta un calendario de las principales tareas del proyecto. Como se ha mencionado, el proceso iterativo e incremental de RUP está caracterizado por la realización en paralelo de todas las disciplinas de desarrollo a lo largo del proyecto, con lo cual la mayoría de los artefactos son generados muy tempranamente en el proyecto, en mayor o menor grado de acuerdo a la fase e iteración del proyecto.

**Cuadro 28:****Calendario del Proyecto**

<b>Disciplinas / Artefactos generados o modificados</b>	<b>Comienzo</b>	<b>Aprobación</b>
<b>FASE DE INICIO</b>		
<b>Modelado del Negocio</b>		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	6/07/10 - 12/07/10	21/09/10 - 25/09/10
<b>Requisitos</b>		
Glosario, Visión, Modelo de Casos de Uso	13/07/10-19/07/10	21/09/10-25/09/10
<b>Análisis / Diseño</b>		
Modelo de Análisis / Diseño	20/07/10-26/07/10	27/07/10-02/08/10
Modelo de Datos	27/07/10-02/08/10	03/08/10-09/08/10
<b>Implementación</b>		
Prototipos de Interfaces de Usuario	03/08/10-09/08/10	10/08/10-16/08/10
Modelo de Implementación	10/08/10-16/08/10	13/07/10-19/07/10
<b>Pruebas</b>		
Casos de Pruebas Funcionales	13/07/10-19/07/10	17/08/10-23/08/10
<b>Despliegue</b>		
Modelo de Despliegue	17/08/10-23/08/10	siguiente fase
<b>Gestión de Cambios y Configuración</b>	durante todo el proyecto	durante todo el proyecto
<b>Gestión del proyecto</b>		
Plan de Desarrollo del Software	6/07/10-12/07/10	21/09/10-25/09/10
<b>Ambiente</b>	6/07/10 - 25/09/10	6/07/10 - 25/09/10

<b>FASE DE ELABORACIÓN</b>		
<b>Modelado del Negocio</b>		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	6/07/10-12/07/10	21/09/10-25/09/10
<b>Requisitos</b>		
Modelo de Casos de Uso del Negocio y Modelo de Objetos del Negocio	13/07/10-19/07/10	21/09/10-25/09/10
<b>Análisis / Diseño</b>		
Modelo de Análisis / Diseño	20/07/10-26/07/10	Revisar en cada iteración
Modelo de Datos	27/07/10-02/08/10	Revisar en cada iteración
<b>Implementación</b>		
Prototipos de Interfaces de Usuario	03/08/10-09/08/10	Revisar en cada iteración
Modelo de Implementación	10/08/10-16/08/10	Revisar en cada iteración
<b>Pruebas</b>		
Casos de Pruebas Funcionales	13/07/10-19/07/10	Revisar en cada iteración
<b>Despliegue</b>		
Modelo de Despliegue	17/08/10-23/08/10	Revisar en cada iteración
<b>Gestión de Cambios y Configuración</b>	durante todo el proyecto	durante todo el proyecto
<b>Gestión del proyecto</b>		
Plan de Desarrollo del Software en su versión 2.0	6/07/10-12/07/10	Revisar en cada iteración
<b>Ambiente</b>	6/07/10 - 25/09/10	6/07/10 - 25/09/10



#### **4.1.6.3 Seguimiento y control del proyecto**

##### **Gestión de Requisitos**

Los requisitos del sistema son especificados en el artefacto Visión. Cada requisito tendrá una serie de atributos tales como: importancia, estado, iteración donde se implementa, etc. Estos atributos permitirán realizar un efectivo seguimiento de cada requisito. Los cambios en los requisitos serán gestionados mediante Solicitudes de Cambio, las cuales serán evaluadas y distribuidas para asegurar la integridad del sistema y el correcto proceso de gestión de configuración y cambios.

##### **Control de Plazos**

El calendario del proyecto tendrá un seguimiento y evaluación semanal por el jefe de proyecto.

##### **Control de Calidad**

Para la revisión de cada artefacto y su correspondiente garantía de calidad se utilizarán las guías de revisión y checklist (listas de verificación) incluidas en RUP.

Las fallas detectadas en las revisiones tendrán un seguimiento para asegurar la conformidad respecto de la solución de dichas deficiencias

##### **Gestión de Riesgos**

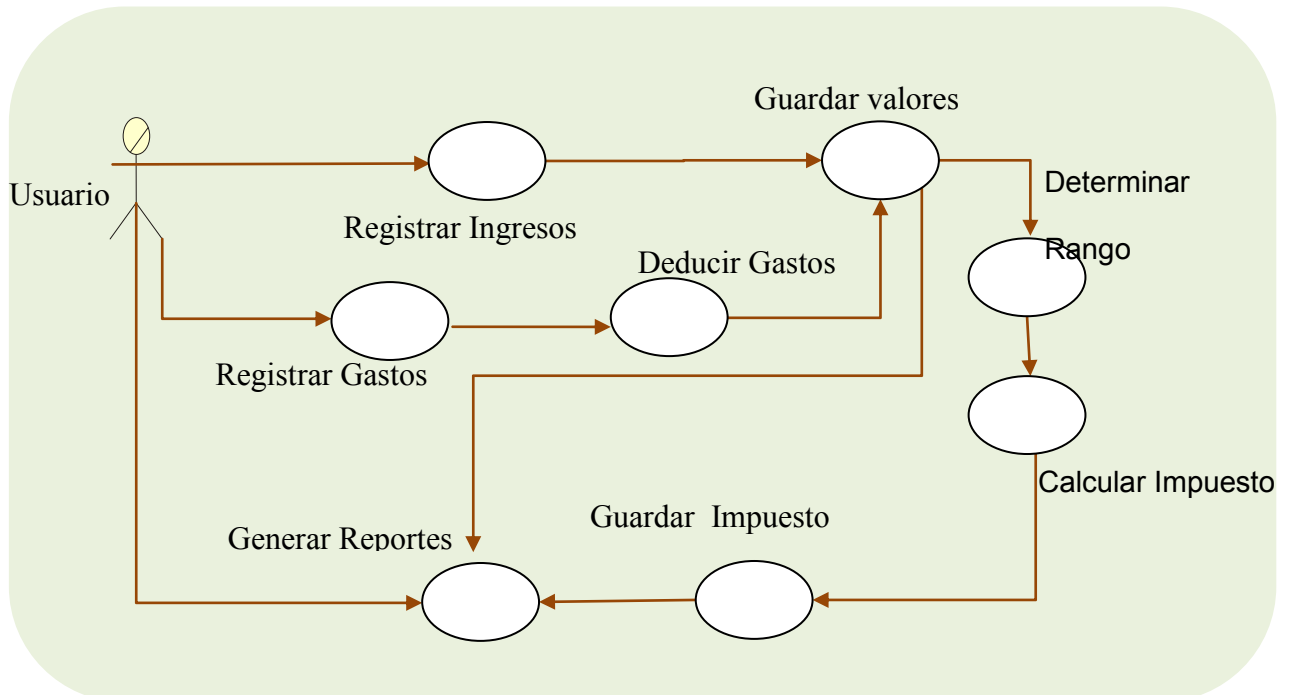
A partir de la fase de Inicio se mantendrá una lista de riesgos asociados al proyecto y de las acciones establecidas como estrategia para mitigarlos o acciones de contingencia. Esta lista será evaluada al menos una vez en cada iteración.

## Gestión de Configuración

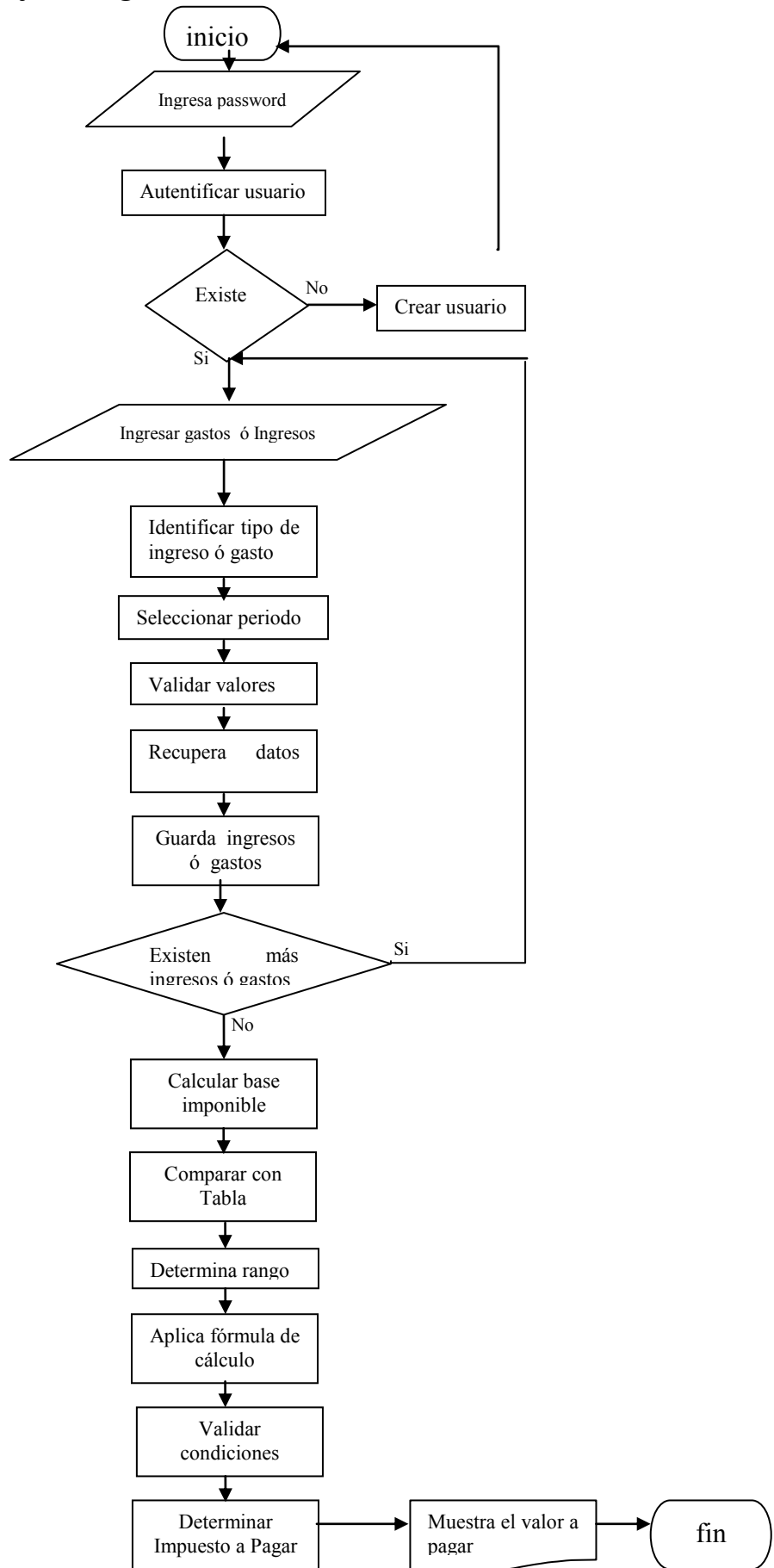
Se creará un registro de los artefactos generados y sus versiones, también se incluirá la gestión de las Solicitudes de Cambio y de las modificaciones que éstas produzcan, informando y publicando dichos cambios para que sean accesibles a todo los participantes en el proyecto.

## 4.2 Modelo de casos de uso del negocio

### 4.2.1 Diagrama general de casos de uso del negocio



#### 4.2.1.1 Diagrama de flujo del negocio



## 4.2.2 Descripción de los casos de uso del negocio

### 4.2.2.1 Registrar ingresos

1.-Proceso de Negocio	Registrar ingresos
2.- Objetivo	Registrar los ingresos en el sistema, para posterior cálculo del impuesto
3.- Actores	Usuario
4.- Precondiciones	
➤	Usuario registrado y dado de alta.
➤	Documento fiscal valido(factura, nota de venta, # secuencial, datos del cliente, etc)
5.- Flujos	
1.-Usuario: ingresa datos	
2.- Sistema: valida datos	
3.- sistema: guarda datos	
4.- Usuario: solicita reporte	
5.- sistema: genera reporte	
6.- Post condiciones	
➤	Actualiza registros de ingresos del usuario.
Excepciones	
➤	En el caso de sueldos y salarios a personas dependientes, con respectivo formulario 107.
➤	Si los datos no son validos el sistema no permite guardar datos

#### 4.2.2.2 Registrar gastos

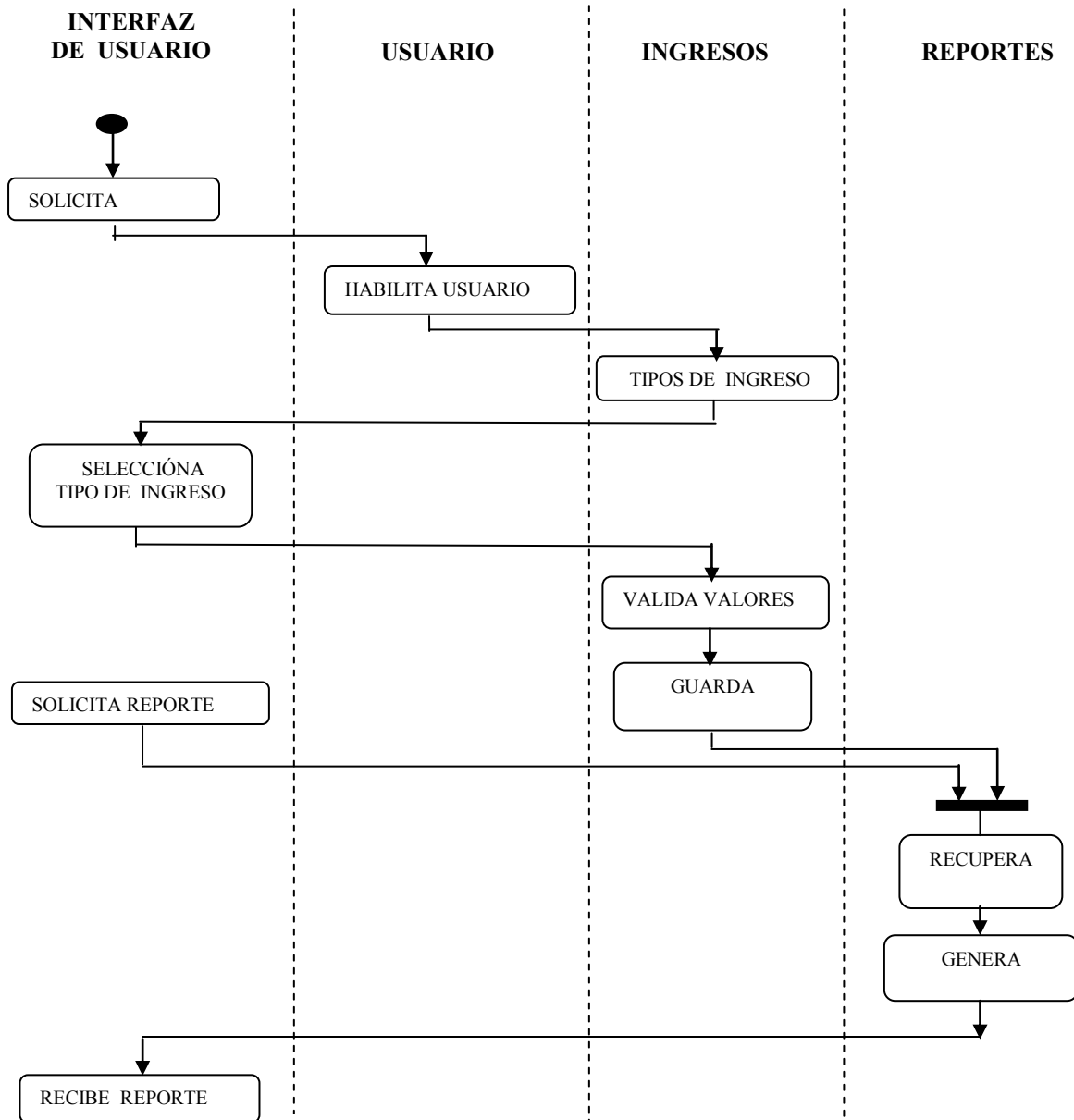
1.-Proceso de Negocio	Registrar gastos
2.- Objetivo	Registrar los egresos en el sistema, para posterior cálculo del impuesto
3.- Actores	Usuario
4.- Precondiciones	
<ul style="list-style-type: none"> <li>➤ Usuario registrado y dado de alta.</li> <li>➤ Documento fiscal valido(factura, nota de venta, # secuencial, datos del cliente, etc)</li> <li>➤ Definir si son bienes o servicios</li> </ul>	
5.- Flujos	
1.-Usuario: ingresa datos 2.- Sistema: valida datos 3.- sistema: guarda datos 4.- Usuario: solicita reporte 5.- sistema: genera reporte	
6.- Post condiciones	
<ul style="list-style-type: none"> <li>➤ Actualiza registros de gastos del usuario.</li> </ul>	
Excepciones	
<ul style="list-style-type: none"> <li>➤ Estos registros deben ser debidamente respaldados por los correspondientes comprobantes</li> <li>➤ Si los datos no son validos el sistema no permite guardar datos</li> <li>➤ Se deberá ingresar el concepto de la transacción con su respectivo valor e IVA que conste en el comprobante.</li> </ul>	

### 4.2.2.3 Calcular Impuesto a la Renta

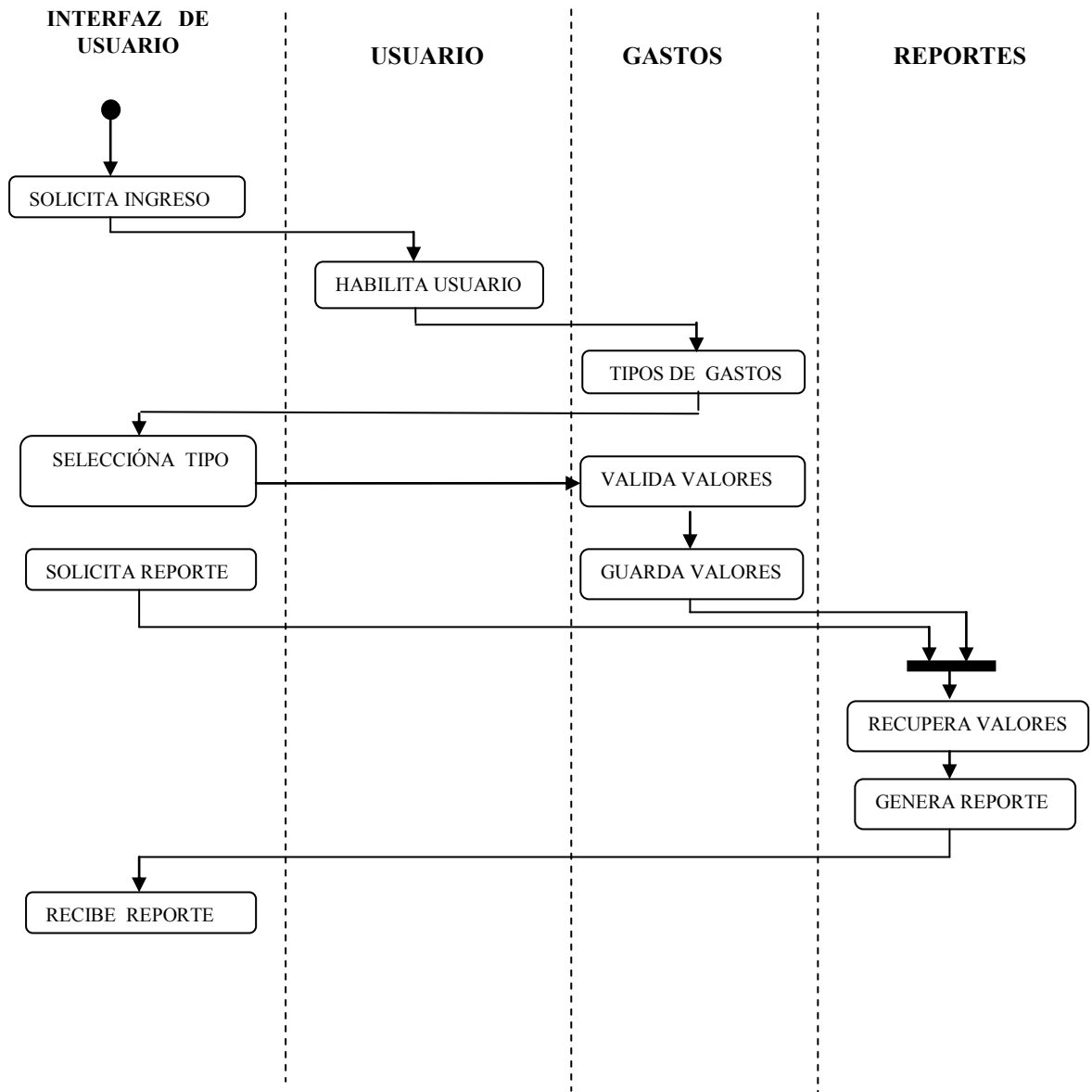
1.-Proceso de Negocio	Calcular impuesto a la Renta
2.- Objetivo	Realizar el cálculo del impuesto a la renta
3.- Actores	Usuario
4.- Precondiciones	
➤	Usuario registrado y dado de alta.
➤	Debe estar registrado ingresos
➤	Debe estar registrado gastos
5.- Flujos	
1.-Usuario:	Solicita calculo
2.- Sistema:	Recupera valores tanto ingresos como egresos
3.- Sistema:	valida valores
4.- sistema:	calcula impuesto
5.- Usuario:	Solicita reporte
6.- sistema:	Genera reporte
6.- Post condiciones	
➤	Calcular el valor del impuesto a pagar.
➤	Determinar los valores de los campos del formulario 102A.
➤	Actualizar reportes del Impuesto generado.
Excepciones	
➤	En caso en que los egresos sean mayores que los ingresos, es decir si se obtiene un valor negativo se toma como cero,
➤	Se calcula interés por mora y multas cuando el contribuyente no ha pagado el valor del impuesto a la renta en los plazos previstos.
➤	Los dividendos y utilidades distribuidas luego del cálculo del impuesto
➤	Los exonerados en convenios internacionales
➤	Los intereses percibidos por depósitos de ahorro a la vista pagados por entidades del sistema financiero nacional
➤	Las prestaciones otorgadas por el IESS, ISSFA, ISSPOL y las pensiones del Estado
➤	Los provenientes de premios de loterías o sorteos auspiciados por la Junta de Beneficencia de Guayaquil y por Fe y Alegría
➤	Los viáticos que se conceden a empleados y funcionarios de las instituciones del Estado, el rancho de los miembros de las Fuerzas Armadas.

### 4.2.3 Diagramas de actividad

#### Diagrama de actividad del Caso de uso: Registrar ingresos

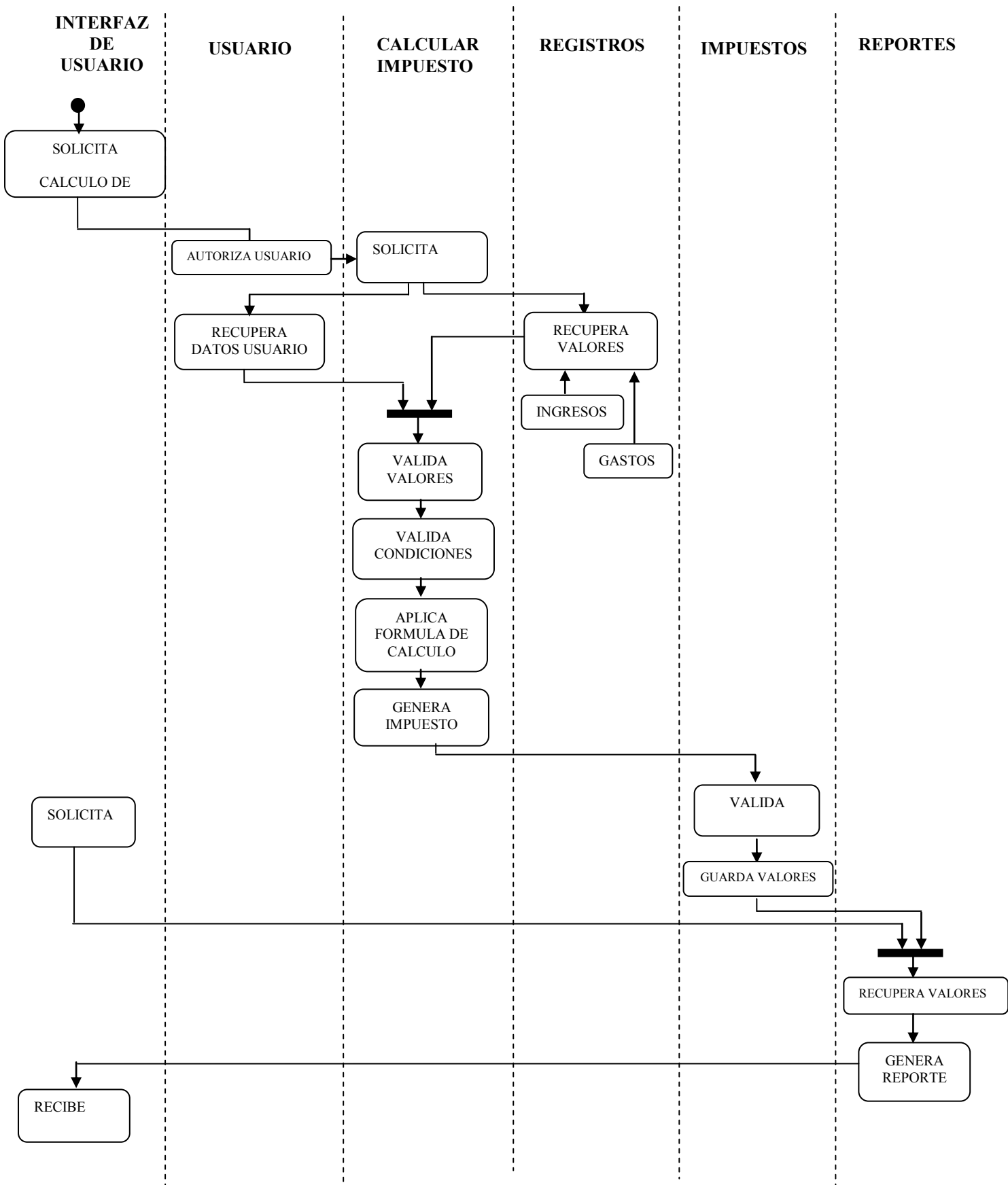


### Diagrama de actividad del Caso de uso: Registrar gastos





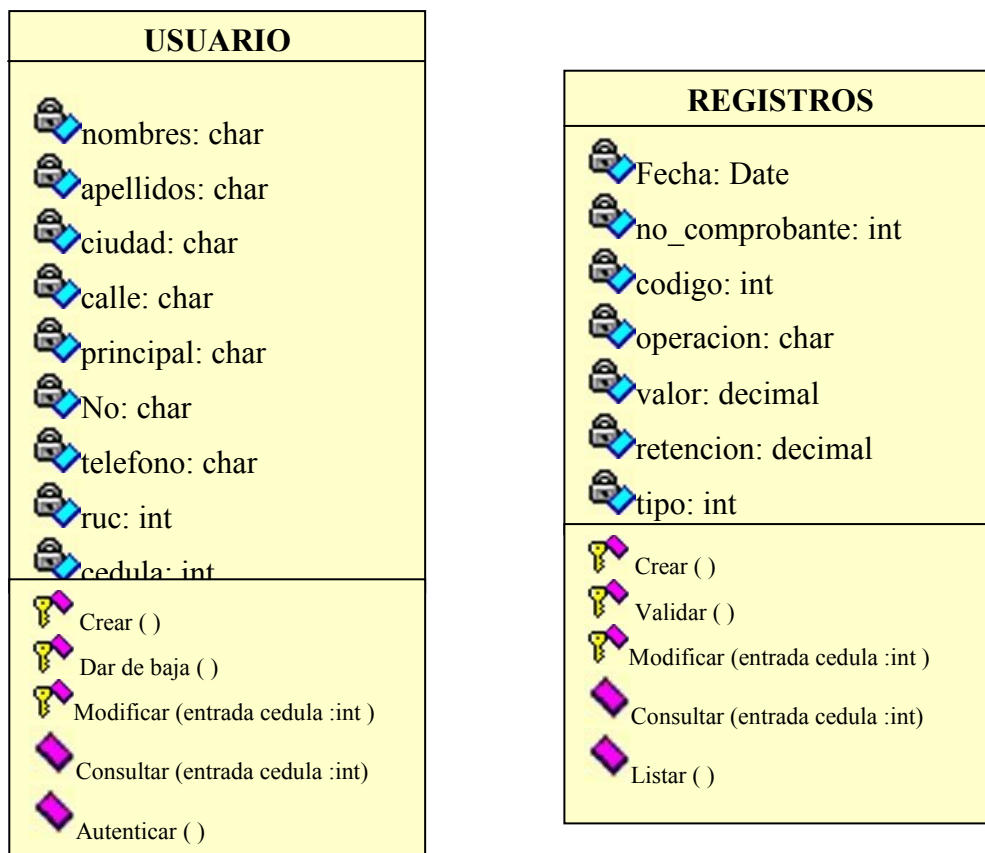
**Diagrama de actividad del Caso de uso: Calcular Impuesto a la Renta**



#### 4.2.4 Diagrama de clases del negocio

Cabe indicar que los diagramas de clase que se presentan a continuación son tomados tal cual como aparecen en la programación, razón por la cual no presentan tildes dado que en programación no se utilizan tildes.

##### 4.2.4.1 Clases


























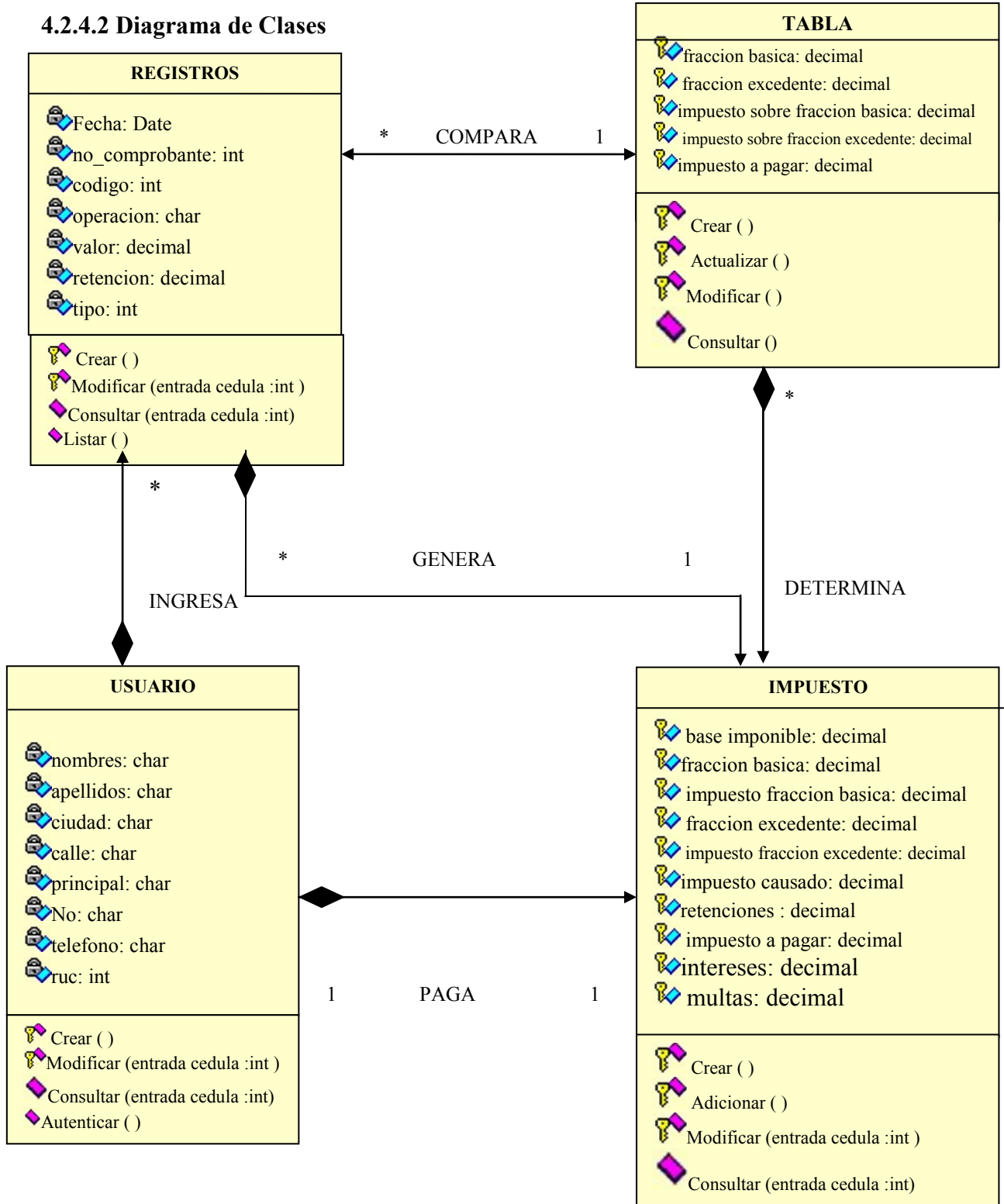
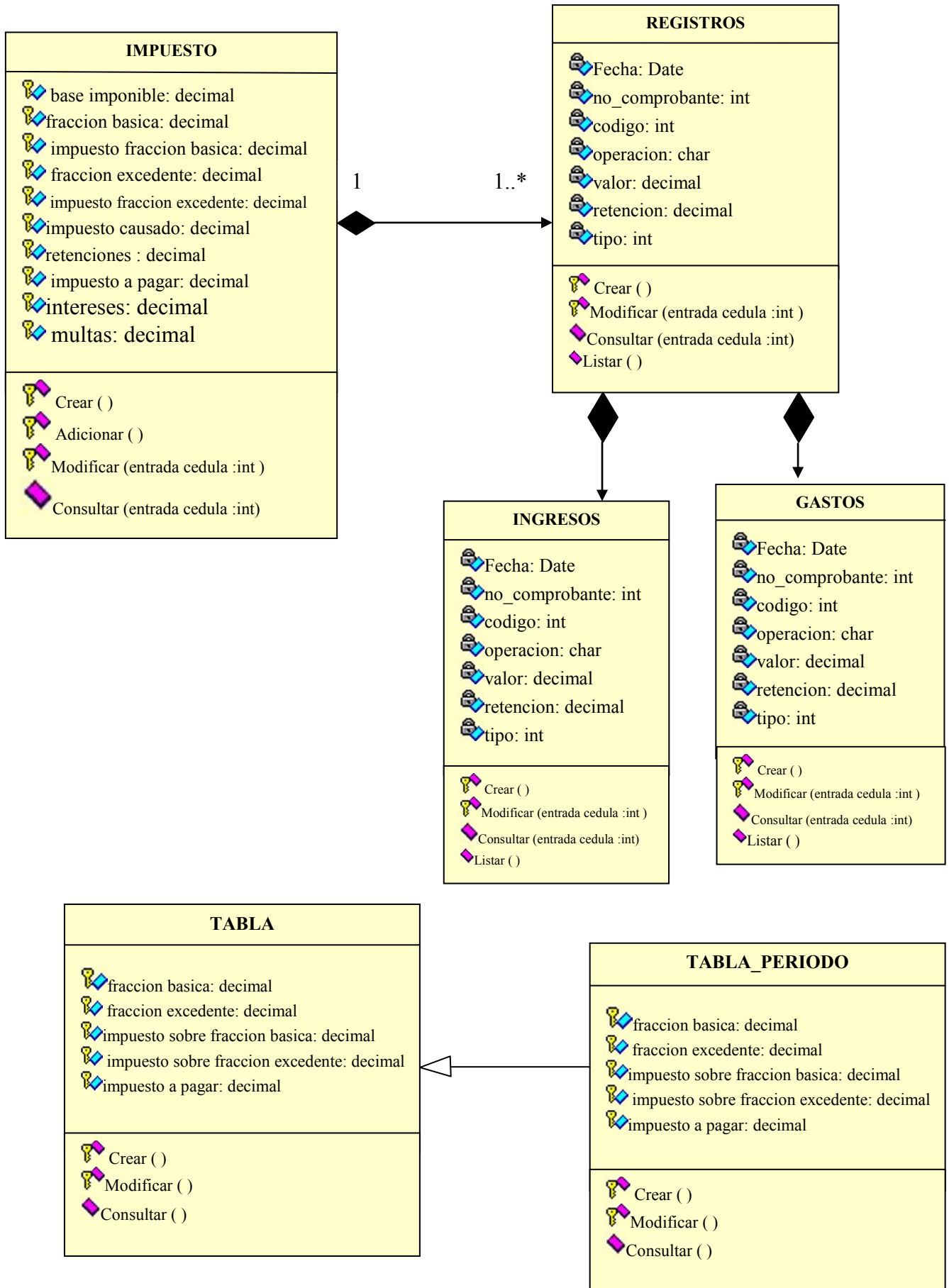
IMPUESTO	
	base imponible: decimal
	fraccion basica: decimal
	impuesto fraccion basica: decimal
	fraccion excedente: decimal
	impuesto fraccion excedente: decimal
	impuesto causado: decimal
	retenciones : decimal
	impuesto a pagar: decimal
	intereses: decimal
	multas: decimal
	Crear ( )
	Adicionar ( )
	Modificar (entrada cedula :int )
	Consultar (entrada cedula :int)

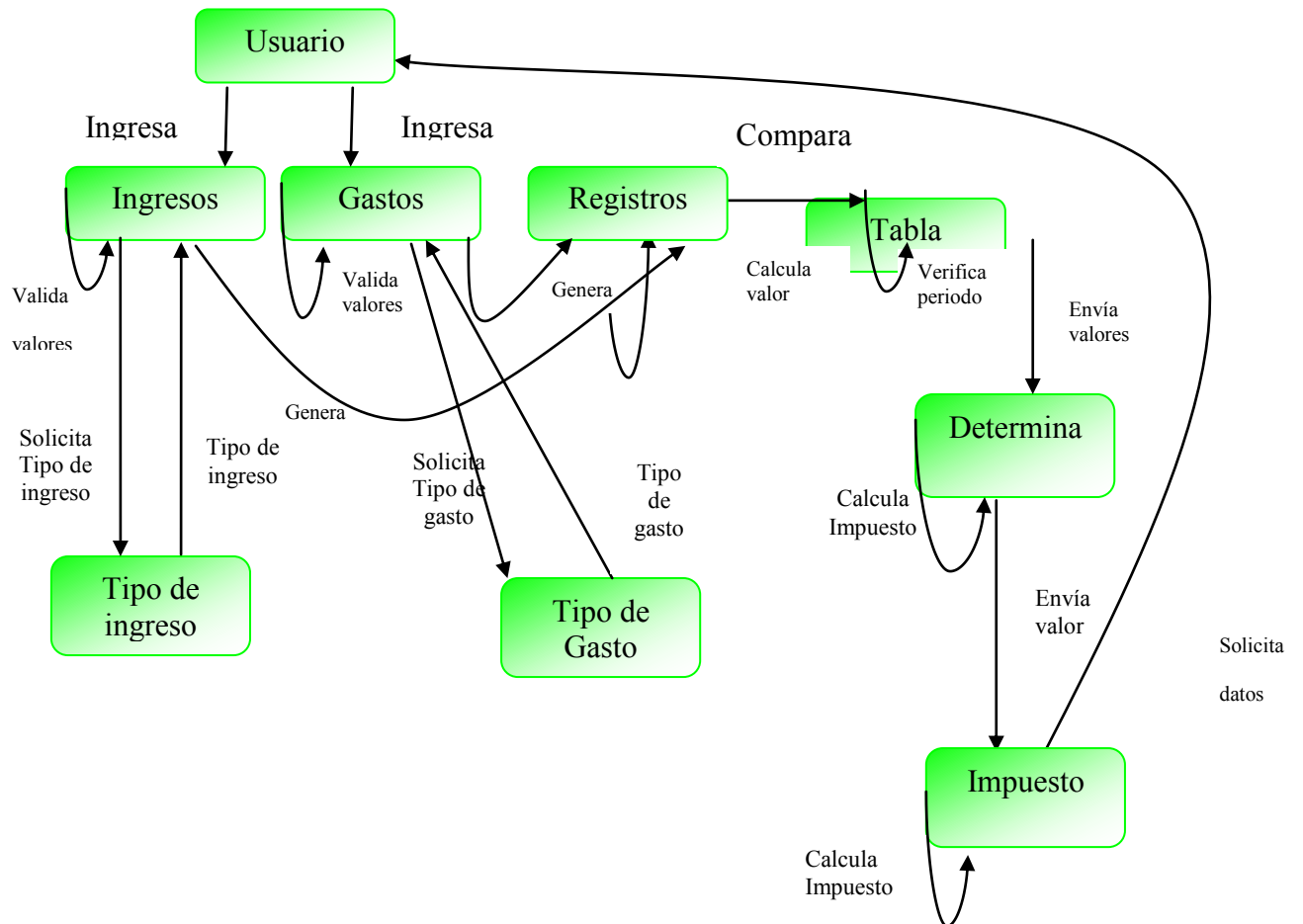
TABLA	
	fraccion basica: decimal
	fraccion excedente: decimal
	impuesto sobre fraccion basica: decimal
	impuesto sobre fraccion excedente: decimal
	impuesto a pagar: decimal
	Crear ( )
	Actualizar ( )
	Modificar ( )
	Consultar ( )

4.2.4.2 Diagrama de Clases

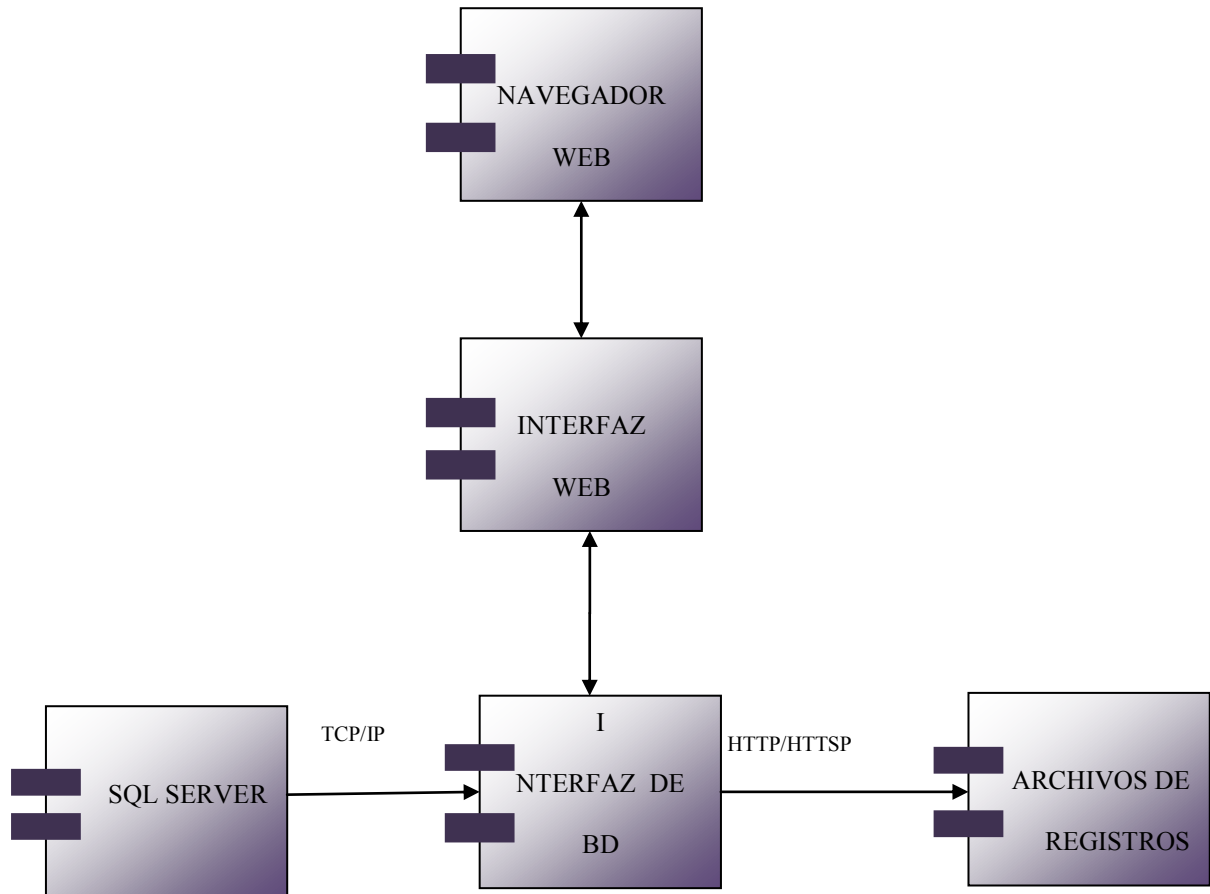




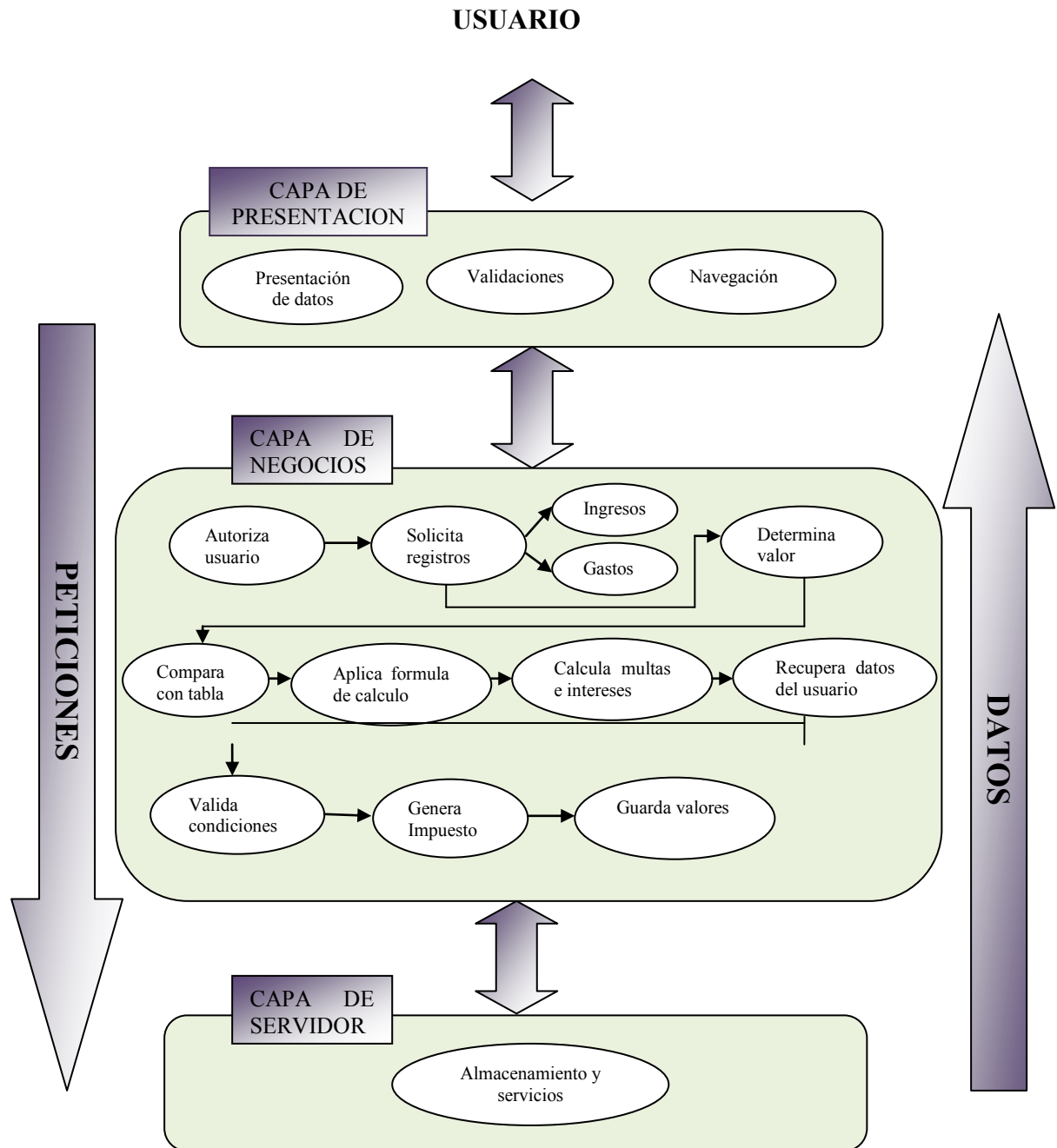
## 4.2.5 Diagrama dinámico



#### 4.2.6 Diagrama de componentes



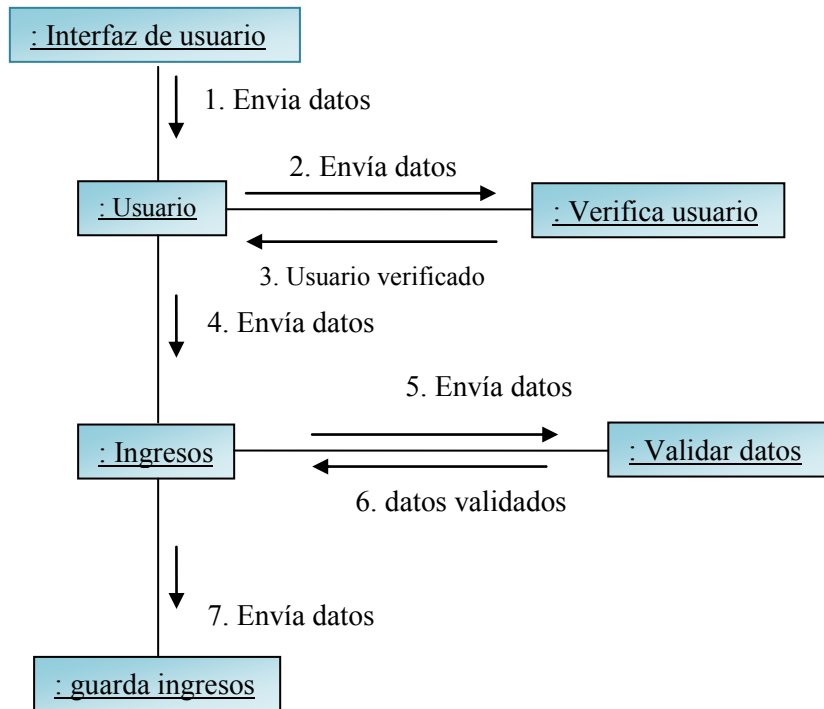
## 4.2.7 Diagrama arquitectónico



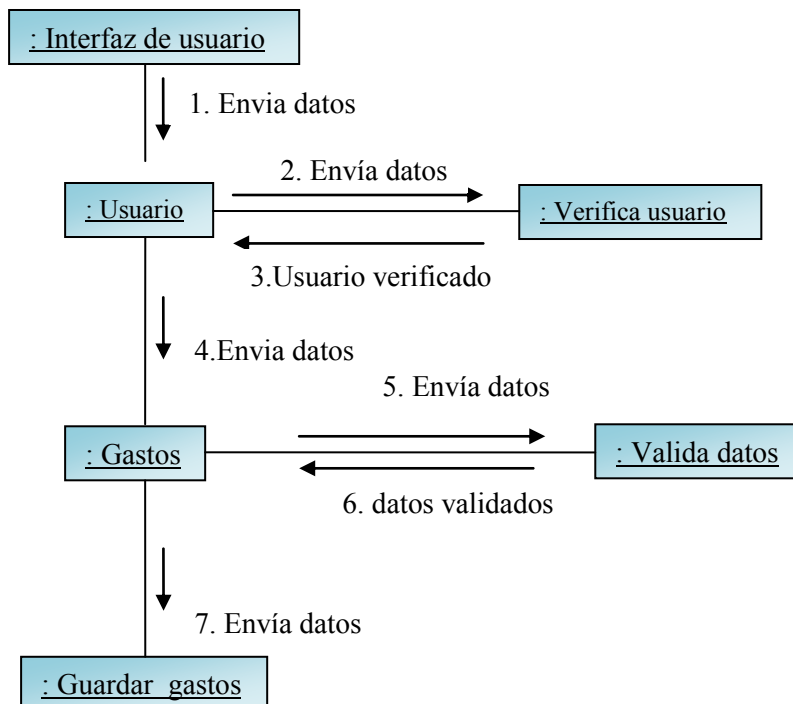


#### 4.2.8 Diagrama de colaboración

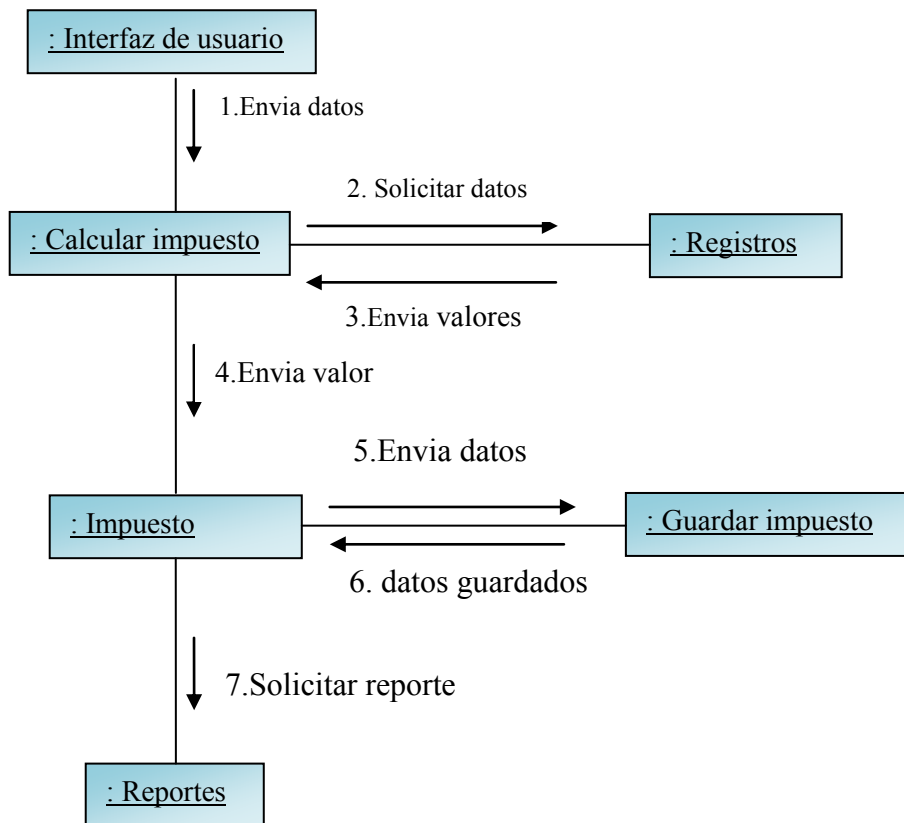
##### Registrar ingresos



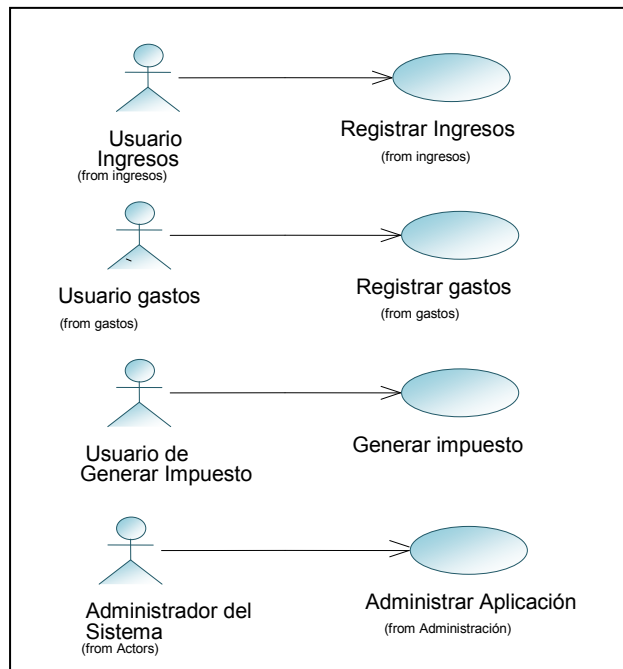
##### Registrar gastos



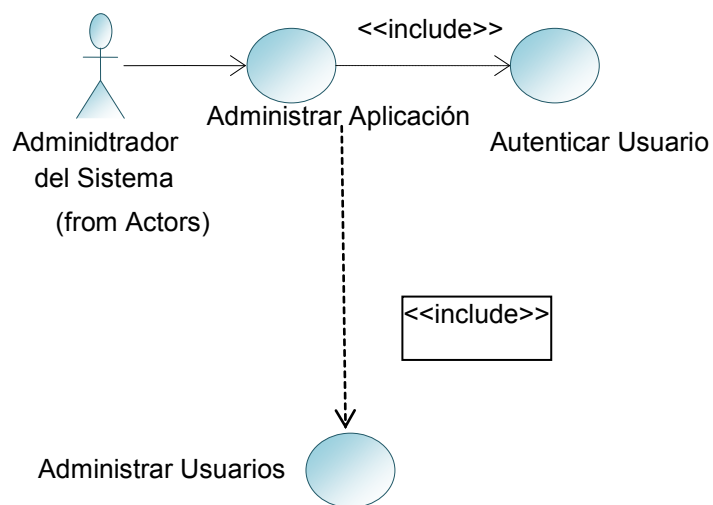
## Calcular impuesto a la renta

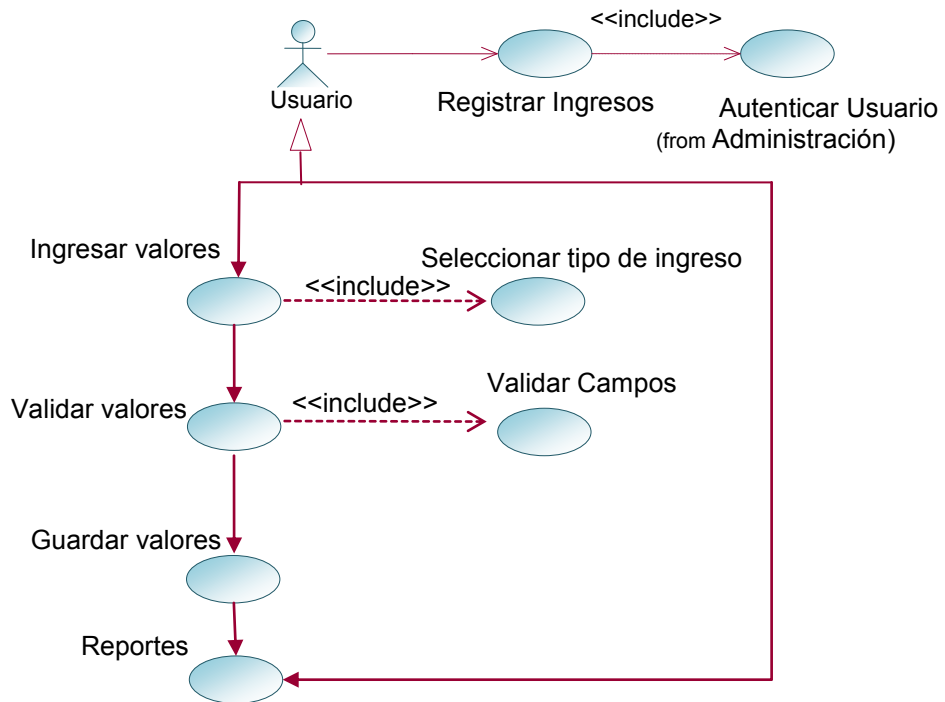


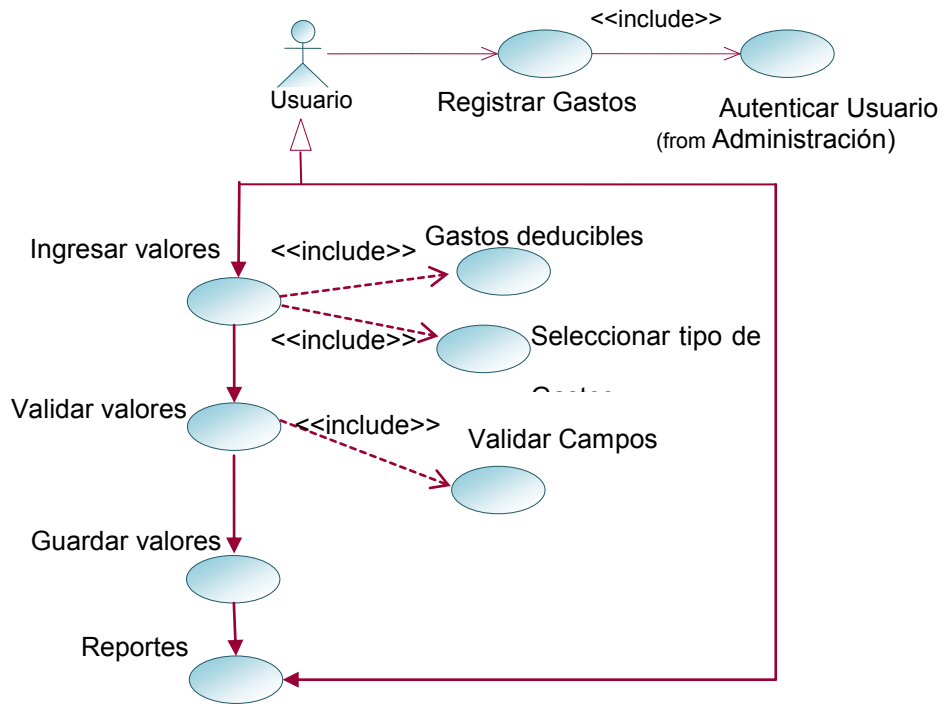
### 4.3 Modelo de casos de uso



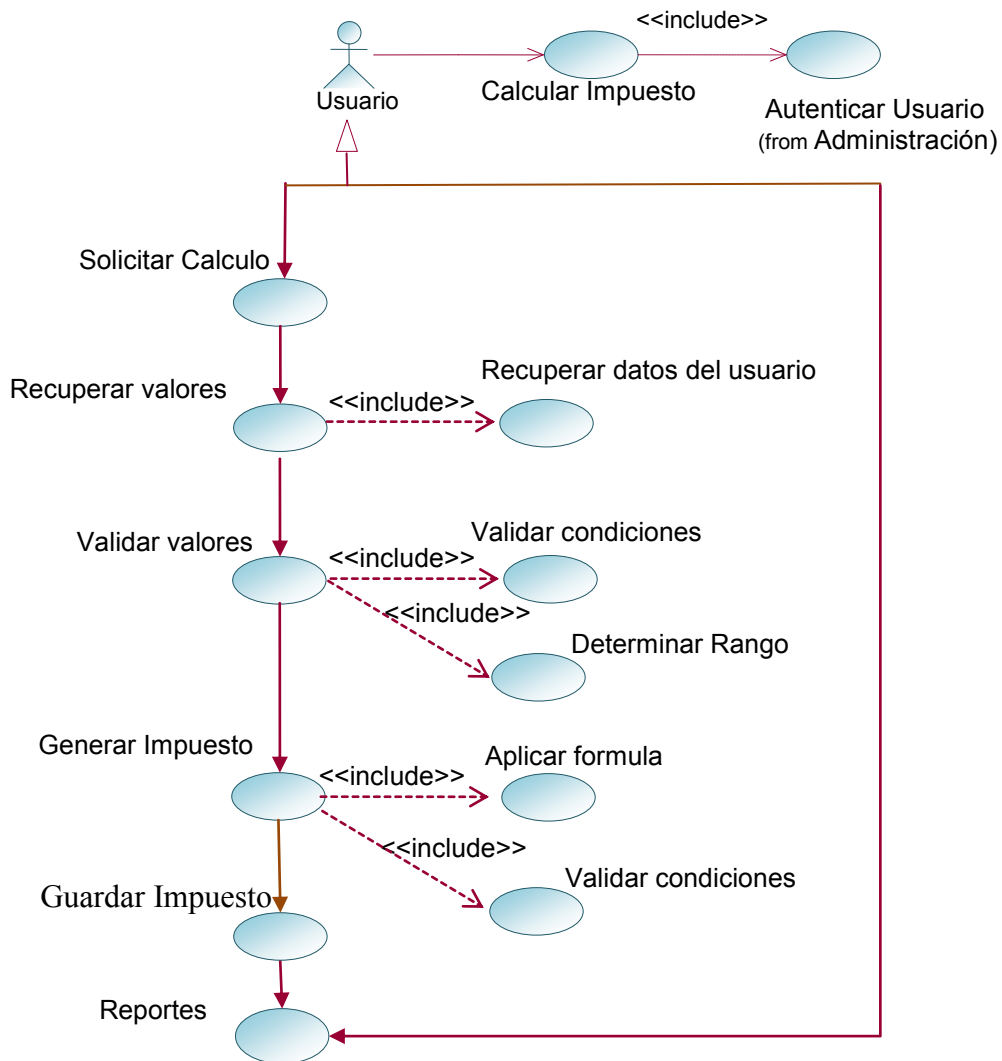
### Diagrama de casos de uso: Administración



**Caso de uso: Registrar ingresos****Caso de uso: Registrar gastos**



**Caso de uso:** Calcular Impuesto a la Renta



## 4.4 GLOSARIO

### *Introducción*

Este glosario recoge todos los términos utilizados en el proyecto de desarrollo del *Sistema*

*de Ingresos y Egresos para el Cálculo del Impuesto a la Renta para Personas Naturales No Obligadas a Llevar Contabilidad.* En este documento se encuentran datos y definiciones de la nomenclatura que se maneja, con el objetivo de estandarizar los términos para todo el proyecto

### ***Propósito***




El propósito principal del glosario es servir como diccionario para todos los usuarios es decir si un término no es entendido con claridad basta con buscar en este documento para aclarar los conceptos manejados en el proyecto.

### ***Alcance***

El alcance del presente documento llega a todos los usuarios del sistema, de tal manera que se utiliza siempre los mismos términos a lo largo del proyecto.

### ***Referencias***

El presente glosario hace referencia a los siguientes documentos:

-  Documento Plan de Desarrollo Software del Proyecto.
-  Documento Visión del Proyecto.
-  Documento Modelado del negocio

### ***Organización del Glosario***

Está organizado por definiciones de términos ordenados de forma ascendente según la ordenación alfabética tradicional del español.

**Cálculo del impuesto a la renta:** Proceso vital en el cual determina el valor del impuesto a cancelar.

**Impuesto:** impuestos - pagos obligatorios de dinero que exige el Estado al individuo.

**Registro de ingresos:** Archivo maestro en el cual se encuentran todos los ingresos que el usuario ha generado.

**Registros de gastos:** Archivo maestro en el cual se encuentran todos los valores o gastos que el usuario a tenido.

**Stakeholders:** Persona o personas que son parte activa de un proyecto.

## 4.5. VISIÓN

### 4.5.1 Introducción

#### *Propósito*

El documento Visión tiene como objetivo recoger todas las necesidades y requisitos que necesita el sistema para garantizar un óptimo funcionamiento y por lo tanto cumplir con todas las necesidades del usuario.

El documento está basado en los requerimientos necesarios para el cálculo del Impuesto a la Renta de personas naturales y la funcionalidad que este debe presentar para el usuario del software, de tal manera que cumpla con los requisitos legales y con las condiciones



solicitadas por el usuario.

### ***Alcance***

El documento Visión tiene la tarea de gestionar el procedimiento para el cálculo del Impuesto a la Renta de personas naturales no obligadas a llevar contabilidad, cuyo desarrollo está encargado el tecnólogo Rafael Chumbi.

Este sistema permitirá al usuario llevar un registro de ingresos clasificados en diferentes tipos al igual que un registro de egresos, permitiéndole el cálculo del valor exacto a pagar por concepto de Impuesto a la Renta

### ***Definiciones, Acrónimos, y Abreviaciones***

**SRI.-** Servicio de Rentas Internas

**IR.-** Impuesto a la Renta

**CAE.-** Corporación Aduanera del Ecuador

**RUP.-** Rational Unified Process. Se trata de una metodología para describir el proceso de desarrollo de software.

**CLR.-** Common Language Runtime, Entorno Común de Ejecución para Lenguajes

**CLI.-** Infraestructura común de lenguajes

**UML** Lenguaje de Modelo Unificado

#### 4.5.2 Posicionamiento

##### *Oportunidad de Negocio*

El sistema permitirá calcular su Impuesto a la Renta a una persona natural no obligada a llevar contabilidad.

Atendiendo al artículo 35 del Reglamento de la Ley de Régimen Tributario interno en el cual codifica a las personas naturales que realicen actividades empresariales y que operen con un capital propio menor a \$60.000,00 al primero de enero del año en curso u obtenga ingresos inferiores a \$100.000,00 o que sus costos o gastos sean menores a \$80.000.00 al 31 de diciembre del año anterior; así como los profesionales, comisionistas, artesanos, agentes, representantes y demás trabajadores autónomos no están obligados a llevar contabilidad.

Este segmento de contribuyentes llevarán una cuenta de ingresos y egresos para determinar su renta imponible, con el propósito de realizar sus declaraciones de Impuesto a la Renta en calidad de personas naturales no obligadas a llevar contabilidad, utilizando para el efecto el formulario SRI-102. Este sistema permite definir en forma clara sus cuentas de ingresos y egresos resultando de fácil comprensión la utilización del mismo para el usuario final.

#### **Cuadro 29**

##### **Sentencia que define el problema**

El problema de	Controlar cuentas de ingresos Controlar cuentas de egresos
----------------	---

	Pagar fuera de los plazos establecidos Pagar multas por mora
Afecta a	Usuario final
El impacto asociado es:	Guardar todos los valores por concepto de ingresos y egresos de tal forma que el usuario pueda determinar su renta imponible, con el propósito de realizar sus declaraciones de Impuesto, lo cual sería muy difícil sin contar con esta herramienta informática.
Una adecuada solución sería	Automatizar el proceso de cálculo para lo cual el sistema debe contar con interfaces amigables y sencillas con el objetivo de obtener su renta imponible.

**Fuente:** Rafael Chumbi

### **Cuadro 30**

#### **Sentencia que define la posición del Producto**

Para	Personas naturales no obligadas a llevar contabilidad.
Quienes	Determinan su renta imponible, con el propósito de realizar sus declaraciones de Impuesto a la Renta en calidad de personas naturales no obligadas a llevar contabilidad.
El nombre del producto	Cálculo del Impuesto a la Renta en calidad de personas naturales no obligadas a llevar contabilidad.
Que	Almacena las cuentas necesarias para determinar el Impuesto a la Renta utilizando una herramienta informática.
Diferenciación	Ventaja Comparativa vs. el sistema actual suministrado por el SRI. Opciones para el usuario que permiten su fácil aplicación y entendimiento para el ingreso de datos y determinación del impuesto.
Nuestro producto	Calcula el Impuesto a la Renta de forma clara y precisa utilizando una interfaz gráfica sencilla y amigable, permitiéndole tener acceso rápido y actualizado a la información requerida con el consecuente ahorro de dinero y tiempo.

**Fuente:** Rafael Chumbi

#### **4.5.3 Descripción de stakeholders (Participantes en el Proyecto) y usuarios**

Para garantizar un producto que satisfaga todos los requerimientos es necesario identificar a los usuarios del sistema y que estos determinen el conjunto de requerimientos necesarios

del proyecto. A continuación se muestra un perfil de los participantes y de los usuarios involucrados en el proyecto, así como los problemas potenciales que se presentan para llegar a determinar la solución.

**Cuadro 31**

**Resumen de Stakeholders**

<b>Nombre</b>	<b>Descripción</b>	<b>Responsabilidades</b>
Mag. Gina Cuadrado	Representante de todos los usuarios potenciales del software.	Representa a todos los usuarios posibles del sistema. Seguimiento del desarrollo del proyecto. Aprueba requisitos y funcionalidades.
Teg. Rafael Chumbi	Tecnólogo en análisis y diseño de sistemas informáticos.	Desarrollar el proyecto Dirigir el proyecto
Ing. Leopoldo Pauta	Ingeniero en Sistemas informáticos.	Control del proyecto

**Fuente:** Rafael Chumbi

**Cuadro 32**

### Resumen de Usuarios

Nombre	Descripción	Stakeholder
Usuario	Es responsable del ingreso de sus datos personales, así como registrar los valores en las respectivas cuentas para posteriormente solicitar el cálculo del Impuesto a la Renta.	Mag. Gina Cuadrado

**Fuente:** Rafael Chumbi

#### *Entorno de usuario*

El usuario puede ingresar al sistema ingresando usuario y clave el mismo que corre sobre el sistema operativo Windows XP, la aplicación tiene la apariencia de cualquier otra página web, resultando para el usuario algo familiar y nada complicado el ingreso al sistema.

### Perfil de los Stakeholders

Representante	Mag. Gina Cuadrado
Descripción	Representante de todos los usuarios potenciales del software.
Tipo	Experta en tributación y contabilidad.
Responsabilidades	Seguimiento del desarrollo del proyecto. Aprueba requisitos y funcionalidades.
Criterio de Éxito	Aprobar todas las etapas del desarrollo del proyecto
Grado de participación	Revisión de requerimientos, cumplimiento de normas y leyes
Comentarios	Ninguno
Representante	Rafael Chumbi
Descripción	Tecnólogo en análisis y diseño de sistemas.
Tipo	Análisis y diseño de sistemas informáticos.
Responsabilidades	Encargado de desarrollar el sistema de ingresos y egresos para determinar el Impuesto a la Renta de personas naturales no obligadas a llevar contabilidad.
Criterio de Éxito	Desarrollar todos los entregables del sistema
Grado de participación	Total.
Comentarios	Ninguno
Representante	Ing. Leopoldo Pauta
Descripción	Representante de la Universidad Israel.
Tipo	Experto sistemas informáticos
Responsabilidades	Encargado de controlar el desarrollo del proyecto.
Criterio de Éxito	Controlar todos los hitos del proyecto.
Grado de participación	Revisión de requerimientos, estructura del sistema
Comentarios	Ninguno

**Fuente:** Rafael Chumbi

### Cuadro 34

### Perfiles de Usuario

Representante	Usuario
Descripción	Cualquier usuario que use el sistema.
Tipo	Usuario del sistema
Responsabilidades	Ingresar información del usuario, valores de ingresos, valores de egresos, así como consultar información.
Criterio de Éxito	Ingresar información correcta así como obtener reportes ágiles y precisos.
Grado de participación	Alta
Comentarios	Ninguno

**Fuente:** Rafael Chumbi

#### 4.5.4 Descripción global del producto

##### **Perspectiva del producto**

Los resultados esperados de la automatización del cálculo del Impuesto a la Renta de Personas Naturales no obligadas a llevar contabilidad, serán medidos en cuanto a: fácil uso, ahorro de tiempo y seguridad.

**Fácil uso:** el sistema solicita el ingreso de datos de una forma sencilla mediante la presentación de pantallas que orientan o sugieren la información que debe ser ingresada.

**Ahorro de tiempo:** Considerablemente se ahorrará tiempo tanto en el ingreso de los datos como en el cálculo del impuesto en vista de que están predeterminadas de antemano todas las variables necesarias para el cálculo por lo tanto este es automático y no manual.

**Seguridad:** Al contar el sistema con un amplio componente legal desde el punto de vista tributario, tenemos la seguridad de que el cálculo del impuesto cumple con las exigencias legales.

### Resumen de características

Los beneficios que obtendrá el cliente con la utilización de este software son:

**Cuadro 35**

#### Resumen de características

Beneficio del cliente	Características que lo apoyan
Rápido acceso a consultas	Al ser una aplicación web se puede acceder fácilmente desde cualquier lugar.
Mayor facilidad para la gestión de usuarios	Base de datos centralizada con la información de todos los usuarios.
Registros de Ingresos	Sistema automatizado de reportes de ingresos.
Registro de Egresos	Sistema automatizado de reportes de egresos.
Control de multas e intereses	El sistema se encarga de generar valores por multas e intereses.
Automatización del Cálculo del Impuesto	El sistema calcula automáticamente el Impuesto a pagar o el saldo a favor.

**Fuente:** Rafael Chumbi

#### 4.5.5 Otros requisitos del producto



**Cuadro 36****Requisitos de Sistema**

<b>Descripción</b>	<b>Prioridad</b>
<b>Uso</b>	
El sistema debe guiar al usuario mediante un menú amigable, dinámico, descriptivo e informativo.	Media
El Control de los usuarios del Sistema, deben autenticarse.	Alta
<b>Confiability</b>	
Los valores generados cumplen con los requisitos legales.	Alta
Seguridad en el Manejo de la Información, la información debe pasar desde la capa del cliente hasta el servidor.	Alta
<b>Mantenimiento</b>	
La solución debe desarrollarse utilizando instrucciones de base de datos SQL Server 2005.	Alta
Se debe manejar nomenclatura estandarizada para las líneas de código.	Alta
<b>Interfaces</b>	
El usuario debe tener instalado Windows XP navegadores: Internet Explorer 6 o más.	Alta
Seguridad. Solamente podrán ingresar usuarios autenticados.	Alta
Las contraseñas deberán tener 8 caracteres alfanuméricos.	Alta

**Fuente:** Rafael Chumbi

**Cuadro 37****Requisitos de Desempeño**

Requerimiento	Descripción	Prioridad
Asignar los montos de acuerdo a lo establecido con la resolución que norma la institución en los procesos de Transferencia.	Una vez registrada la información se deben generar las variables establecidas por defecto.	Alta
Mostrar la información registrada a través de consultas o informes en cualquier momento.	Visualizar los informes respectivos para la toma de dediciones de la gerencia.	Alta
Contar con los cálculos exactos y confiables en los informes a generar.	Visualizar informes sin errores de cálculo que son vitales para la toma de decisiones	Alta
Tener un control en cada campo de las interfaces.	Controlar el acceso de datos que no correspondan y se evite el ingreso de datos errados.	Alta

Fuente: Rafael Chumbi

### Cuadro 38

#### Requisitos de Entorno

La solución será desarrollada en Visual Web Developer Express.	Alta
Los controles del usuario serán creados de acuerdo a los requerimientos del propio proceso.	Media
El servidor mínimo Pentium IV, Windows XP.	Alta

Fuente: Rafael Chumbi

## 4.6 Especificación de casos de uso

### Cuadro 39

#### Especificación del caso de uso: Registrar ingresos

PROPÓSITO	PRECONDICIONES	ACTIVACIÓN	FLUJO PRINCIPAL	VARIACIONES	EXCEPCIONES
Registrar los ingresos en el sistema, para calcular posteriormente el impuesto	Usuario habilitado Documento fiscal válido (factura, nota de venta, # secuencial, datos del cliente, etc)	A discreción de un usuario habilitado	1.-Usuario: ingresa datos 2.- Sistema: valida datos 3.- Sistema: guarda datos 4.- Usuario: solicita reporte 5.- sistema: Genera reporte	Si los datos no son válidos el sistema no permite guardar datos	En el caso de sueldos y salarios a personas dependientes, con su respectivo formulario 107.

**Fuente:** Rafael Chumbi

### Cuadro 40

#### Especificación del caso de uso: Registrar gastos

PROPÓSITO	PRECONDICIONES	ACTIVACIÓN	FLUJO PRINCIPAL	VARIACIONES	EXCEPCIONES
Registrar los egresos en el sistema, para posterior cálculo del impuesto	Usuario habilitado Documento fiscal válido (factura, nota de venta, # secuencial, datos del cliente, etc) Definir si son bienes o servicios	A discreción de un usuario habilitado	1.-Usuario: ingresa datos 2.- Sistema: valida datos 3.- sistema: guarda datos 4.- Usuario: Solicita reporte 5.- sistema: Genera reporte	-Si los datos no son válidos el sistema no permite guardar datos -Se deberá ingresar el concepto de la transacción con su respectivo valor e IVA que conste en el comprobante.	Estos registros deben ser debidamente respaldados por los correspondientes comprobantes.

**Fuente:** Rafael Chumbi

### Cuadro 41

#### Especificación del caso de uso: Calcular Impuesto a la Renta

PROPÓSITO	PRECONDICIONES	ACTIVACIÓN	FLUJO PRINCIPAL	VARIACIONES	EXCEPCIONES
Realizar el cálculo del impuesto a la renta	Usuario habilitado Registrado ingresos Registrado gastos	A discreción de un usuario habilitado	1.-Usuario: ingresa datos 2.- Sistema: valida datos 3.- sistema: calcula impuesto 4.- Usuario: Solicita reporte 5.- sistema: Genera reporte	En caso de que los egresos sean mayores que los ingresos, se obtiene un valor negativo quedando como cero. Se calcula interés por mora y multas cuando el contribuyente no ha pagado el valor del impuesto a la renta en los plazos previstos.	Los dividendos y utilidades distribuidas luego del cálculo del impuesto Los exonerados en convenios internacionales Los intereses percibidos por depósitos de ahorro a la vista pagados por entidades del sistema financiero nacional Las prestaciones otorgadas por el IESS, ISSFA, ISSPOL y las pensiones del Estado Los provenientes de premios, loterías o sorteos auspiciados por la Junta de Beneficencia de Guayaquil y por Fe y Alegría Los viáticos que se conceden a empleados y funcionarios de las instituciones del Estado, el rancho de los miembros de las Fuerzas Armadas.

**Fuente:** Rafael Chumbi

## 4.7 ESPECIFICACIONES ADICIONALES

### Sistema operativo Windows XP

Primer sistema operativo de Microsoft orientado al consumidor que se construye con un

núcleo y arquitectura de Windows NT y que se encuentra disponible en versiones para PC de 32 y 64 bits.

Posibilidad de unirse a un dominio, en vez de únicamente a grupos de trabajo, y soporte para 2 procesadores (Home a partir del SP para multicore).

A diferencia de sus versiones anteriores presenta mejoras en la estabilidad y de la eficacia. Tiene una Interfaz gráfica de usuario (GUI) perceptiblemente reajustada, un cambio de Microsoft promovido para un uso más fácil que en las versiones anteriores.

*Características:*

- ✚ Secuencias más rápidas de inicio y de hibernación.
- ✚ Capacidad del sistema operativo de: desconectar un dispositivo externo, instalar nuevas aplicaciones y controladores sin necesidad de reiniciar.
- ✚ Una nueva interfaz de uso más fácil, incluyendo herramientas para el desarrollo de temas de escritorio.
- ✚ Uso de varias cuentas, que permite que un usuario guarde el estado actual y las aplicaciones abiertas en su escritorio y permita que otro usuario abra una sesión sin perder esa información.
- ✚ ClearType, diseñado para mejorar legibilidad del texto encendido en pantallas de cristal líquido (LCD) y monitores similares.
- ✚ Escritorio Remoto, que permite a los usuarios abrir una sesión con una computadora que funciona con Windows XP a través de una red o Internet,

teniendo acceso a sus usos, archivos, impresoras, y dispositivos; y

- ✚ Soporte para la mayoría de módems ADSL y conexiones wireless, así como el establecimiento de una red FireWire.

## Requisitos de hardware

	Mínimo	Recomendado	Óptimo
Procesador	233 MHz	300 MHz	500 MHz o más
Memoria	64 MB RAM (funcionamiento limitado)	128 MB RAM	256 RAM o más
Video	Super VGA (800 x 600) 512 KB	SXGA (1280 x 1024) con 8 MB de video	SXGA (1280x1024) con 64 MB de video o más
Espacio en Disco Duro	1.5 GB	17.0 GB	80 GB o más
Unidades	CD-ROM	CD-ROM y DVD-ROM	CD-R/RW y DVD-R/RW DL
Periféricos	Teclado y mouse	Teclado y mouse	Teclado y mouse
Otros (Multimedia)		Tarjeta de Sonido, Altavoces, y/o Auriculares	Tarjeta de Sonido, Altavoces, y Auriculares

## 4.8 PROTOTIPOS DE INTERFAZ DE USUARIO

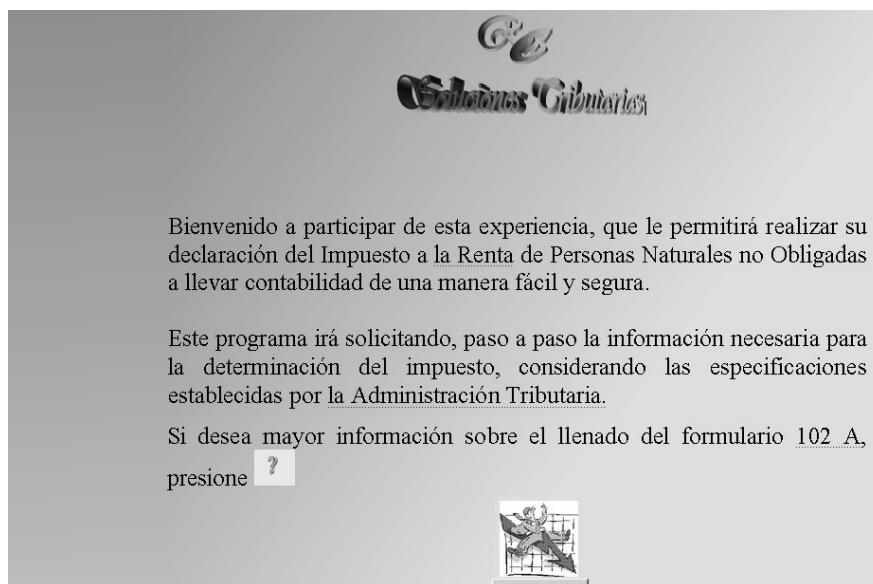
### INICIO

Al ingresar el sistema desplegará la pantalla de inicio, en la que constan datos del producto como: Nombre del Sistema, Autor, Ciudad y Fecha.

Para comenzar dar clic en inicio.

## BIENVENIDA

El Programa da la bienvenida al sistema indicando su misión y dejando la posibilidad de obtener mayor información presionando el icono de ayuda.



## AYUDA

En la pantalla de ayuda se desplegará un documento de cuatro páginas con las instrucciones de llenado del formulario 102 A, con el objetivo de que el usuario tenga un panorama global de la información que el sistema irá solicitando.

**FORMULARIO 102A - INSTRUCTIVO**  
**DECLARACION DEL IMPUESTO A LA RENTA**  
**PERSONAS NATURALES Y SUCESIONES INDIVISAS NO**  
**OBLIGADAS A LLEVAR CONTABILIDAD**

---

**NOTAS IMPORTANTES:**

Los formularios podrán ser llenados a máquina (de escribir o impresora), en forma legible manuscrita (a tinta); o de ambas formas.

Tanto el original como las copias no deberán contener manchones, repisados o enmendaduras.

Todos los valores deben ser ingresados en valor absoluto; es decir, no se deben utilizar ni el signo "menos", ni "paréntesis" para valores negativos.

Se debe utilizar el punto para separar miles y la coma para decimales; siempre se deberán incluir dos decimales (aproximando el segundo). Por ejemplo: Si la cantidad resultante fuera 12.435,501 registre 12.435,50 o si la cantidad resultante fuera 12.435,506 registre 12.435,51.

Todos los campos no utilizados deberán ser anulados con una línea horizontal. No se

Luego de realizar la consulta respectiva, regresa a la pantalla de bienvenida y elige continuar.

**SEGURIDAD**

Inmediatamente solicita el nombre de usuario y contraseña.



The image shows a login interface for 'Soluciones Tributarias'. At the top, there is a logo with the letters 'R' and 'E' and the text 'Soluciones Tributarias'. Below this is a header 'Identificación de Usuario y Clave'. The main area contains the instruction 'Ingrese su identificación y clave'. On the left, there is an image of a key. On the right, there are two input fields: 'USUARIO' with the text 'rafael' and 'CLAVE' with four dots. Below the fields is an 'ACEPTAR' button.



## MANTENIMIENTO DE USUARIOS

Al ingresar con éxito el nombre de usuario y contraseña dar clic en aceptar. La siguiente pantalla en aparecer es la de mantenimiento de usuarios, en la cual podemos crear nuevos usuarios y modificar o consultar los existentes.



## BÚSQUEDA

En esta pantalla puede realizar búsquedas por períodos y por nombres de contribuyentes.

Dando clic en Select ingresa a la pantalla de ingreso de datos.

SELECT	NOMBRE	APELLIDO	CIUDAD	CALLE	NUM TELEFONO	RUC	CEDULA
>>>	RAFAEL	CHUMBI MENDIETA	CUENCA	DUITAMA	P1 2898545	0101885309001	0101885309
>>>	GINA PATRICIA	CUADRADO SANCHEZ	CUENCA	LARGA	13-45 2839003	0102414091001	0102414091
>>>	LUZ MARIA	SANCHEZ VELEZ	CUENCA	DUITAMA	1 2898243	0601221740001	0601221740
>>>	EDISSON HERNAN	CALLE AMBROSI	AMBATO	LIBERTAD	1850 2809558	0601228521001	0601228521
>>>	LUIS	LOPEZ RODRIGUEZ	QUITO	AV. AMAZONAS 654	258956	1803129595001	1803129595

## INGRESO DE DATOS

El ingreso de datos está programado en dos ejes de trabajo: ingresos y egresos.

Los ingresos y los egresos están codificados de acuerdo a los códigos establecidos en el formulario 102 A de declaración de Impuesto a la Renta para Personas Naturales no Obligadas a Llevar Contabilidad.

[REGRESAR AL MENU](#)  
[GENERAR BASE IMPONIBLE](#)

TIPO DE INGRESOS

ELEGIR TIPO	TIPO	DESCRIPCION
--->	481	ACTIVIDAD EMPRESARIAL
--->	502	ARRIENDO DE INMUEBLES
--->	503	ARRIENDO DE ACTIVOS
--->	611	LIBRE EJERCICIO DE PROFESION

ELEGIR CODIGO_TIPO	CODIGO_TIPO	DESCRIPCION
--->	501	PREDIOS AGRICOLAS

TIPO DE EGRESOS

ELEGIR TIPO_EG	TIPO_EG	DESCRIPCION
--->	491	ACTIVIDAD EMPRESARIAL
--->	621	LIBRE EJERCICIO DE PROFESION

ELEGIR TIPOS_EG	TIPOS_EG	DESCRIPCION
--->	522	ARRIENDO DE INMUEBLES
--->	523	ARRIENDO DE ACTIVOS

ELEGIR TIPOS_EG	TIPOS_EG	DESCRIPCION
--->	622	SUELDOS SALARIOS

ELEGIR TIPOS_EG	TIPOS_EG	DESCRIPCION
--->	110	CEDULA DEL CONYUGUE

DATOS INFORMATIVOS

ELEGIR COD_TIPO	COD_TIPO	DESCRIPCION
--->	701	PREMIOS DE LOTERIAS,RIFAS,APUESTAS
--->	711	NO DE FORMULARIO O COMPROBANTE
--->	721	VALOR IMPUESTO PAGADO
--->	702	HERENCIAS, LEGADOS Y DONACIONES

TERCERA EDAD DISCAPACIDAD

## TIPO DE INGRESOS

*Actividad Empresarial – Arriendo de Inmuebles – Arriendo de Activos – Libre Ejercicio de la Profesión*

Aquí se detallan los ingresos obtenidos por el contribuyente por concepto de Actividad Empresarial, Arriendo de Inmuebles, Arriendo de Activos y Libre Ejercicio de La Profesión, por los cuales tiene los respectivo comprobantes: Facturas, Notas de Venta, Planillas de IESS, etc.

La información solicitada es:

- ✚ Fecha: fecha que consta en la factura o nota de venta.
- ✚ No. de Comprobante: el número de la factura, nota de venta u otro.
- ✚ Código: el código será de generación automática, por lo tanto no es necesario ingresar dato alguno.
- ✚ Operación: constarán las palabras bienes o servicios según el caso.
- ✚ Valor sin IVA: para el cálculo del Impuesto a la Renta se colocarán los subtotales de las facturas sin incluir el IVA.
- ✚ Retención: las retenciones son pagos anticipados del Impuesto a la Renta y son efectuadas por personas obligadas a llevar contabilidad, sociedades, empresas estatales y contribuyentes especiales. Aquí se ingresará el valor de retención efectuada al documento registrado.

En esta pantalla podemos crear, agregar, modificar, eliminar, seleccionar y totalizar registros.

Una vez generados los ingresos regresamos al menú principal para continuar el registro de datos.

FECHA

No. comp.

CODIGO

OPERACION

VALOR SIN IVA

RETENCION

NUMERO DE CEDULA  
0102414091

481

NUEVO AGREGAR MODIFICAR ELIMINAR TOTAL REGRESAR

INGRESOS

SELECT tipos	fecha	No_com	codigo	operacion	valor	ret	cedula
>>>	481	2005/04/02	120	1	BIEN	12550.0	0102414091

## PREDIOS AGRÍCOLAS

El cálculo de la renta imponible por predios agrícolas es un valor único del 5% del valor del avalúo del predio; ingresar el valor del avalúo, dar clic en guardar y el cálculo será automático.

Dar clic en regresar para retornar al menú principal.

**PREDIOS AGRICOLAS**

ELEGIR	cedula	tipo cod_tipo	descripcion	valor	retencion	periodo
-->>	0102414091	501 1	PREDIOS AGRICOLAS	400	0	2005

INGRESE EL VALOR QUE CONSTA EN EL PREDIO

REGRESAR

GUARDAR

NUEVO

ELIMINAR

MODIFICAR

TOTAL

## SUELDOS, SALARIOS E INDEMNIZACIONES

Ingresar los valores de sueldos, salarios e indemnizaciones laborales por concepto de ingresos bajo dependencia, que consten en el formulario 107 entregado por su empleador, así como la retención que consta en dicho formulario.

Guardar la información y regresar al menú principal.

ELEGIR	CEDULA	TIPO	COD_TIPO	DESCRIPCION
-->>	0102414091	612	2	SUELDOS SALARIOS,INDEMNIZACIONES



612  
0102414091

SUELDOS  
SALARIOS,INDEMNIZACIONES

RETENCION

REGRESAR

NUEVO  
GUARDAR  
ELIMINAR  
MODIFICAR  
TOTAL

## PREMIOS, LOTERIAS, RIFAS, APUESTAS

Colocar el valor del premio obtenido y la respectiva retención, guardar y regresar.

ELEGIR	CEDULA	TIPO	COD_TIPO	DESCRIPCION
-->>	0102414091	701	5	PREMIOS DE LOTERIAS,RIFAS,APUESTAS

701  
0102414091


TOTAL\_TIPO  
1332332

PREMIOS DE  
LOTERIAS,RIFAS,APUESTAS

RETENCION

REGRESAR

NUEVO  
GUARDAR  
ELIMINAR  
MODIFICAR  
TOTAL



## TIPO DE EGRESOS

### ACTIVIDAD EMPRESARIAL – LIBRE EJERCICIO DE LA PROFESIÓN

En esta sección colocar los egresos y deducciones establecidas en la Ley de Régimen Tributario Interno, por ejemplo:

Gastos con comprobantes de venta: honorarios, materia prima, suministros y materiales, repuestos y herramientas, activos fijos, patentes, promoción y publicidad, arrendamiento, servicios, combustibles, servicios básicos etc.

481

FECHA  NUMERO DE CEDULA

NUMERO DE COMPROBANTE  ARRIENDO

CODIGO  SERVICIOS BASICOS

OPERACION BIEN HONORARIOS

VALOR SIN IVA  SUMINISTROS

INTERESES POR CREDITO

OPERACIONES ▾

CON FACTURA ▾

SIN FACTURA ▾

REGRESAR

Button NUEVO AGREGAR MODIFICAR ELIMINAR TOTAL

Gastos sin comprobantes de venta: depreciaciones, impuestos, primas de seguros, pérdidas por caso fortuito, sueldos.

481

FECHA 22/05/2005 0:00:00 NUMERO DE CEDULA 0102414091

NUMERO DE COMPROBANTE 1212

CODIGO 5

OPERACION BIEN

VALOR SIN IVA 1225

OPERACIONES ▾

CON FACTURA ▾

SIN FACTURA ▾

DEPRECIACIONES

IMPUESTOS, CONTRIBUCIONE

PRIMAS SEGUROS

PERDIDA CASO FORTUITO

SUELDOS

Nuevo elemento

REGRESAR

Button NUEVO AGREGAR MODIFICAR ELIMINAR TOTAL

EGRESOS

SELECT tipos_eg	fecha	No_com	codigo	operacion	valor	cedula
>>>	491	22/05/2005 0:00:00	1212	5	BIEN	1225 0102414091

## ARRENDAMIENTO DE INMUEBLES – ARRENDAMIENTO DE ACTIVOS

Los gastos en esta sección son limitados a ciertos conceptos, razón por la cual se establece los datos requeridos mediante la barra de desplazamiento.

En Arrendamiento de Inmuebles y Activos Fijos el programa calcula automáticamente el valor de la depreciación y de mantenimiento establecidos por ley.

Para el caso de mantenimiento se necesita generar los gastos de mantenimiento dando clic en EGRESOS POR MANTENIMIENTO, en vista de que se puede tener en gastos sustentados por facturas un valor mayor o menor al 1% del avalúo que establece la ley debiendo en el primer caso seleccionar el 1% del avalúo y en el segundo caso el valor menor de mantenimiento, aspecto que el programa lo hace automáticamente.

0102414091

**522**

FECHA: 01/01/1900 0:00:00 DEPRECIACION:

NUMERO DE COMPROBANTE: TOTAL MANTENIMIENTO:

CODIGO: 3

OPERACION: PAGO IMPUESTO PRE

VALOR: 2500

GUARDAR

NUEVO MODIFICAR TOTAL REGRESAR

AGREGAR ELIMINAR EGRESOS POR MANTENIMIENTO

ELEGIR tipos_eg	fecha	No_com	codigo	operacion	valor	cedula
-->>	522	01/01/1900 0:00:00	3	DEPRECIACION	2500	0102414091
-->>	522	01/01/1900 0:00:00	4	MANTENIMIENTO 0		0102414091

## SUELDOS Y SALARIOS

Colocar el valor del aporte personal al IESS que consta en el formulario 107

**CÉDULA DEL CONYUGUE** Colocar como dato informativo la cédula del cónyuge para realizar la deducción de impuestos por concepto de sociedad conyugal.

Esta deducción se realizará siempre y cuando los cónyuges tengan un negocio que administren juntos y que no tengan ingresos por libre ejercicio de la profesión o por sueldos y salarios.

ELEGIR	CEDULA	CED_CONYUGE	TIPOS_EG	CODIGO	DESCRIPC
-->>	0102414091	0101885291	110	7	CEDULA DEL C

cedula del conyuge

REGRESAR

GUARDAR

ELIMINAR

MODIFICAR

### TERCERA EDAD Y DISCAPACIDAD

Escoger Tercera Edad si el contribuyente tiene 65 años cumplidos al primero de enero del año al que pertenece la declaración. Esta opción le permitirá hacer una deducción del doble de la fracción básica.

Discapacidad será escogido por personas con discapacidad que tengan el respectivo carné de CONADIS, esta opción le permitirá hacer una deducción del triple de la fracción básica.

<input checked="" type="checkbox"/> TERCERA EDAD	DESCUENTO POR TERCERA EDAD 14800	DESCUENTO TOTAL 37000
<input checked="" type="checkbox"/> DISCAPACIDAD	DESCUENTO POR DISCAPACIDAD 22200	
<input type="button" value="ACEPTAR"/> <input type="button" value="REGRESAR"/>		

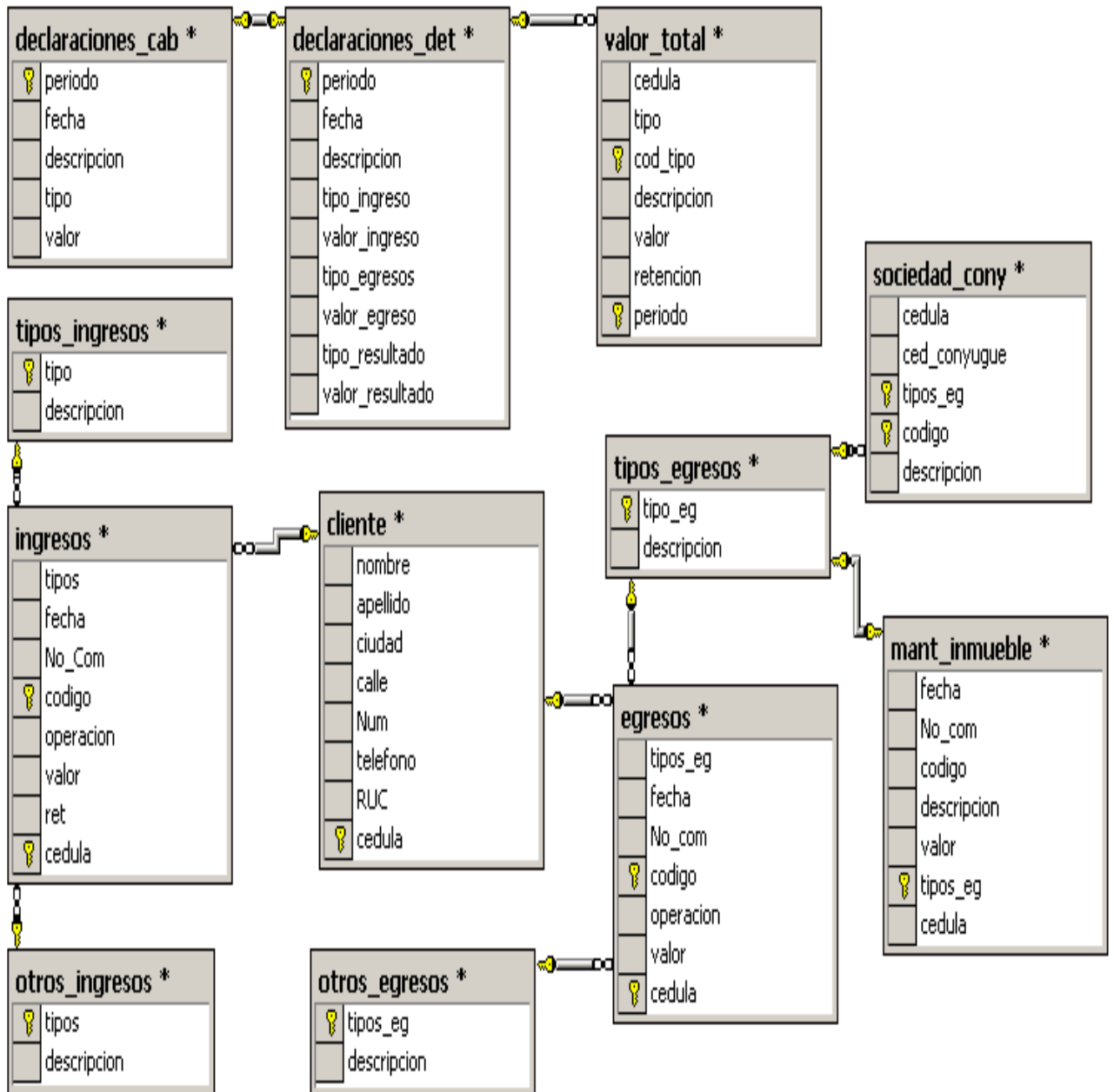


## CÁLCULO DEL IMPUESTO

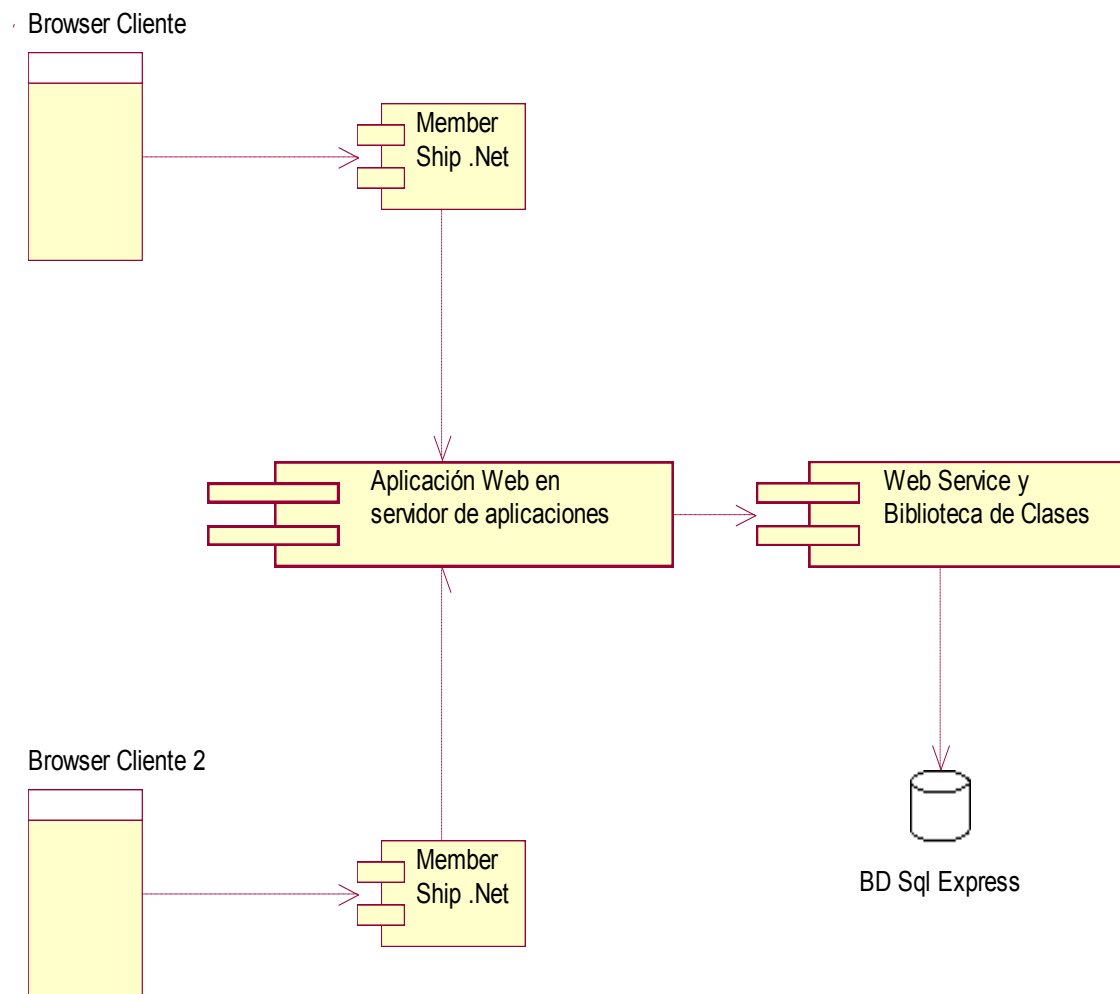
Finalmente genera el Impuesto a la Renta recopilando todos los datos solicitados anteriormente, mostrando en pantalla el Formulario 102 A, de esta forma tenemos el formulario lleno como debería ser presentado para la pertinente declaración.

SRI Servicio de Rentas Internas		DECLARACION DEL IMPUESTO A LA RENTA PERSONAS NATURALES Y SUCESIONES INDIVISAS NO OBLIGADAS A LLEVAR CONTABILIDAD		No. [ ]	
FORMULARIO 102 A RESOLUCION N° 2005-0637					
100 IDENTIFICACION DE LA DECLARACION					
102	AÑO	104	N° FORMULARIO QUE SUSTITUYE		
	2005				
106	No. Empleados bajo nómina				
200 IDENTIFICACION DEL CONTRIBUYENTE					
201	RUC, C.I. o Pasaporte	202	DENOMINACION O APELLIDOS Y NOMBRES COMPLETOS	205	NUMERO
	0102414091		CUADRADO SANCHEZ		1:
203	CIUDAD	204	CALLE PRINCIPAL		
	CUENCA		LARGA		
ACTIVIDAD EMPRESARIAL DE PERSONAS NATURALES NO OBLIGADAS A LLEVAR CONTABILIDAD					
ACTIVIDADES EMPRESARIALES CON REGISTROS DE INGRESOS Y EGRESOS		451	INGRESOS	451	GASTOS DEDUCIBLES
			12550		1225
RENTA DE BIENES RAICES Y OTROS ACTIVOS		AVALUO		RENTA IM	
501	PREDIOS AGRICOLAS	501	8000	531	400
502	ARRIENDO DE INMUEBLES	512	0	532	0
503	ARRIENDO DE OTROS ACTIVOS	513	0	533	0
		SUBTOTAL		531 + 532 + 533	400
IDENTIFICACION DEL CONYUGE					
(-) 50% INGRESOS ATRIBUIBLES SOCIEDAD CONYUGAL		110	C.I / PASAPORTE	111	VAL
			101885291		5862,5
RENTA DEL TRABAJO PERSONAL					
LIBRE EJERCICIO DE PROFESION U OCUPACION LIBERAL (INCLUYE PROFESIONALES, COMISIONISTAS, ARTESANOS, AGENTES, REPRESENTANTES Y DEMAS TRABAJADORES AUTONOMOS)		611	0	621	0
SUELDOS, SALARIOS, INDEMNIZACIONES Y OTROS INGRESOS LIQUIDOS DEL TRABAJO EN RELACION DE DEPENDENCIA		612	0	622	0
		SUBTOTAL		631 + 632	0
OTROS INGRESOS					
INGRESOS POR REGALIAS				727	0
OTROS INGRESOS GRAVADOS PROVENIENTES DEL EXTERIOR				728	0
RENDIMIENTOS FINANCIEROS				729	0
VALOR No. FORMULARIO O COMPROMISOS VALOR IMPUESTO PAGADO					
PREMIOS DE LOTERIAS, RIFAS, APUESTAS		701	0	711	0
HERENCIAS, LEGADOS Y DONACIONES		702	0	712	0
				721	
				722	

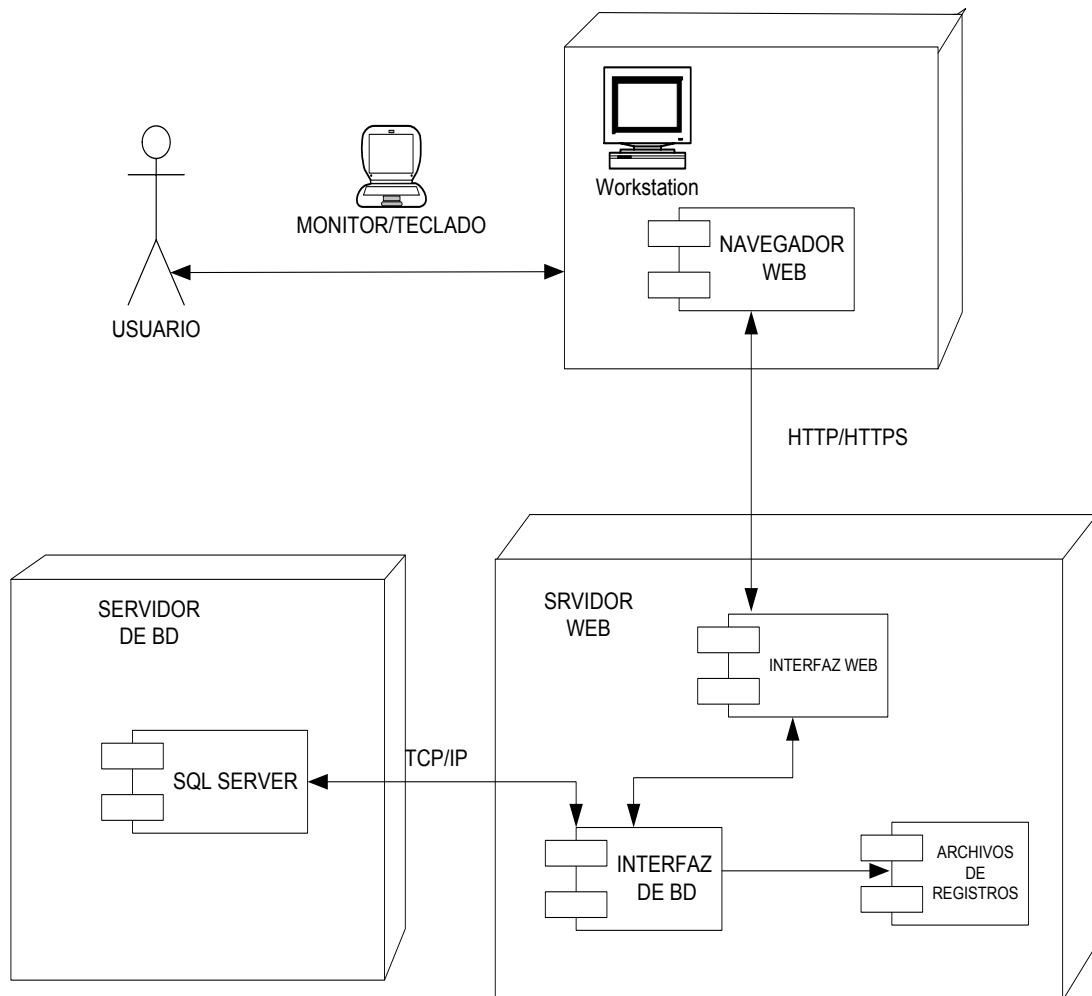
## 4.9 Modelo de Datos



## 4.10 Modelo de Implementación



## 4.11 Modelo de despliegue



## 4.12 CASOS DE PRUEBA

### 4.12.1 Casos de prueba del caso de uso: Registrar ingresos

#### Descripción

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso Registrar ingresos. Las pruebas realizadas a este caso de uso son:

- ✚ Validar valores
- ✚ Validar campos
- ✚ Escoger tipo de ingreso
- ✚ Guardar valores
- ✚ Consultar información
- ✚ Totalizar ingresos por tipo

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

### **Condiciones de ejecución**

El usuario debe autenticarse y escoger el período.

### **Entrada**

Ingresar información para determinar período de declaración del Impuesto.

El sistema muestra la interfaz en la cual permite seleccionar el tipo de ingreso.

El usuario ingresa los valores además la fecha, detalle si es bien o servicio, número de comprobante, retención.

El usuario guarda la información.

El usuario solicita totalizar los ingresos.

**Resultado esperado**

Se espera que los valores de ingresos se guarden correctamente de acuerdo al tipo de ingreso, periodo y usuario.








**Evaluación de la Prueba**

Realizada y satisfactoria

**4.12.2 Casos de prueba del caso de uso: Registrar gastos****Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso Registrar gastos

Las pruebas realizadas a este caso de uso son:

-  Validar valores
-  Validar campos
-  Escoger tipo de gasto
-  Aplicar gastos deducibles
-  Guardar valores
-  Consultar información
-  Totalizar gastos por tipo

**Condiciones de ejecución**

El usuario debe autenticarse.

Deben existir ingresos de acuerdo al tipo de gasto.

**Entrada**

Ingresar a la interfaz de registrar gastos

El sistema muestra un listado con todos los tipos de gastos

El usuario ingresa la información solicitada

El usuario guarda la información.

**Resultado esperado**

Se espera que los valores de gastos se guarden correctamente de acuerdo al tipo de gasto, período y usuario.

**Evaluación de la Prueba**

Realizada y satisfactoria

### **4.12.3 Casos de prueba del caso de uso: Generar Impuesto**

#### **Descripción**

Este caso de uso permite a un usuario generar el Impuesto causado durante el período, además de intereses por mora, tomando en cuenta todas las condiciones legales como son gastos deducibles, cónyuge, discapacidad, tercera edad, etc.

#### **Condiciones de ejecución**

El usuario debe autenticarse.

Deben existir registro de ingresos y gastos

#### **Entrada**

Ingresar a la interfaz de generar Impuesto

El usuario selecciona el período.

El usuario ingresa la información solicitada

El usuario guarda la información.



**Resultado esperado**

Se espera que la información se guarde correctamente y se relacione con un único usuario.

Se espera los valores de los campos para llenar el formulario 102 A.

**Evaluación de la Prueba**

Realizada y satisfactoria

**4.12.4 Casos de prueba del caso de uso: Administrar Aplicación****Descripción**

Este artefacto cubre el conjunto de pruebas realizadas sobre el Caso de Uso Administrar Aplicación.

Las pruebas realizadas a este caso de uso son:

Autenticar Usuario.

Administrar usuarios.

Administrar grupos o perfiles.

El entorno del cual partiremos para realizar la prueba será el formulario de entrada de la aplicación.

#### **4.12.4.1 Autenticar usuario**

##### **Descripción**

Este caso de uso permite a un visitante del sitio identificarse como un usuario válido y le asigna privilegios y restricciones de acuerdo al perfil al que pertenezca.

##### **Condiciones de ejecución**

El visitante debe conectarse con un Browser compatible con Internet Explorer 6.0

##### **Entrada**

El visitante ingresa en el link Iniciar Sesión

El sistema muestra la interfaz de autenticación.

El visitante ingresa el nombre de usuario y la clave de acceso

El sistema valida los datos y da una respuesta.

##### **Resultado esperado**

Se espera que el sistema le dé la bienvenida y conceda los privilegios de su perfil, siempre y cuando los datos del visitante sean correctos; caso contrario se debe mostrar un mensaje de error y el visitante no podrá acceder a ninguna función del sistema.

## **Evaluación de la Prueba**

Realizada y satisfactoria

### **4.12.4.2 Administrar usuarios**

#### **Descripción**

Este caso de uso permite a un administrador crear un usuario con sus respectivas claves de acceso para luego distribuirlas a los usuarios.

#### **Condiciones de ejecución**

El usuario debe autenticarse como administrador.

Debe estar activada la autenticación desde internet para el sitio.

#### **Entrada**

El sistema muestra un listado con todos los usuarios creados con su respectivo perfil.

El usuario puede crear un nuevo usuario, modificar información o dar de baja al mismo.

El usuario selecciona una opción.

El usuario guarda la información.

**Resultado esperado**

Se espera que el registro de usuarios se actualice correctamente con las acciones del administrador.

**Evaluación de la Prueba**

Realizada y satisfactoria

## 5. CONCLUSIONES Y RECOMENDACIONES

### 5.1. CONCLUSIONES

- ✚ En el Ecuador la importancia del Impuesto a la Renta radica en sus niveles de recaudación, los ingresos tributarios por este impuesto representan alrededor del 35% del Presupuesto General del Estado, manteniendo una presión fiscal, que obliga a la Administración Tributaria y a los contribuyentes fijar una especial atención en su aplicabilidad.
- ✚ En la República del Ecuador se realiza una distinción importante en el tratamiento del Impuesto a la Renta, tanto para personas naturales como para las sociedades, las primeras determinan el Impuesto a la Renta mediante cálculos específicos por cada base imponible de la tabla establecida en el Art. 36 de la Ley de Régimen Tributario, la misma que se actualiza anualmente; en tanto que las sociedades aplican un porcentaje único del 25% de Impuesto a la Renta a sus utilidades.
- ✚ Por la complejidad del cálculo el Impuesto a la Renta de las personas naturales no obligadas a llevar contabilidad, es necesario de la presencia de un asesor contable y/o tributario para efectuarlo, labor que sería reemplazada por la herramienta informática propuesta en este proyecto, con el consecuente ahorro de tiempo y

dinero, minimizando el riesgo de errores u omisiones dados por el desconocimiento en la aplicación de la Ley y su Reglamento, o por errores involuntarios en el cálculo.

- ✚ El presente estudio se concentra en el cálculo del Impuesto a la Renta de Personas Naturales no obligadas a llevar contabilidad, en vista de que, este sector está desatendido por los programas informáticos actuales. Las personas naturales no obligadas a llevar contabilidad se limitan a realizar el cálculo de forma manual.
- ✚ El 80% de los encuestados manifestó que desconocían como declarar el IR, el 70% desconoce cuando tiene que pagar por este impuesto, el 76% no sabe cuánto tiene que pagar por IR, el 46% está consciente de la existencia de multas y pago de intereses en declaraciones tardías, el 54% conoce cuando se generan pagos indebidos o en exceso, el 84% desconoce los requisitos para declarar y el 90% indica que es necesario automatizar el proceso y que accedería a la aplicación en la Web.
- ✚ La viabilidad de la creación de este software se fundamenta en el sustento teórico obtenido de las disposiciones legales establecidas en el Código Tributario, la Ley del Régimen Tributario Interno y su Reglamento; y el sustento práctico por su fácil uso, ahorro de tiempo y seguridad.
- ✚ La metodología utilizada es la orientada a objetos, que divide los flujos de trabajo en actividades y tareas; los casos de uso dirigen el proceso de desarrollo de software basado en los requisitos de los usuarios hasta transformarlos en un producto software.

- ✚ Este software además de automatizar el cálculo del Impuesto a la Renta, permite obtener registros digitales de Ingresos y Egresos del contribuyente, requisitos exigidos en el Art. 35 del Reglamento de Aplicación a la Ley Orgánica de Régimen Tributario Interno.
- ✚ Las pruebas realizadas en los principales casos de usos, así como, en los posibles escenarios que se pueden presentar, demuestran las ventajas que el software ofrece al realizar los procesos de cálculo en forma rápida, sencilla y confiable, obteniendo resultados fiables como son: el impuesto causado, valor a pagar, anticipos, multas e intereses, entre otros.
- ✚ El diseño de la página Web se basa en la tecnología Microsoft ASP.NET, ya que permite presentar la información al usuario en cualquier explorador o dispositivo cliente; e implementa la lógica de aplicación mediante el código de la parte servidor.
- ✚ El Proceso Unificado fue el utilizado como marco de desarrollo, por su carácter interactivo e incremental, compuesto de cuatro fases denominadas Inicio, Elaboración, Construcción y Transición. Cada una de estas fases está dividida en una serie de iteraciones que dan como resultado el incremento del producto, estas iteraciones se dividen en Análisis de requisitos, Diseño, Implementación y Prueba.

## 5.2. RECOMENDACIONES

- ✚ Es importante contar con la participación de expertos tanto en la parte informática como en el ámbito de aplicación del sistema, para este caso en el área tributaria, permitiendo obtener un producto final que cumple las disposiciones legales y los requerimientos tecnológicos.
- ✚ Se recomienda la metodología orientada a objetos porque admite que las partes del programa (en este caso objetos) pueden ser reutilizados en la confección de otros programas o implementación de nuevos módulos, reduciendo el tiempo de realización, ganando en claridad, mantenimiento y productividad.
- ✚ Es recomendable la adopción de un diseño de aplicaciones distribuidas, ya que aumenta la reusabilidad, reduce la cantidad de recursos, y los costes necesarios de desarrollo y mantenimiento.
- ✚ Difundir el uso del presente sistema informático como una herramienta de ayuda al contribuyente para la determinación del Impuesto a la Renta, utilizando interfaces sencillas, reduciendo los errores de cálculo y omisiones por desconocimiento de las disposiciones legales.



## BIBLIOGRAFÍA

1. Constitución de la República del Ecuador (2008)
2. Código Tributario de la República del Ecuador vigente al 2008
3. Ley de Propiedad Intelectual Ecuatoriana (2008)
4. Ley del Régimen Tributario Interno, Ecuador (2008)
5. Reglamento para la aplicación del Régimen Tributario Interno, Ecuador (2008)
6. KORTH, Henry. SILVERSCHATZ, Abraham. SUDARSHAN, S. 1998. Fundamentos de Bases de Datos. McGraw-Hill. Tercera Edición.
1. FOWLER Martin, KENDALL, Sccott, "UML Gota a Gota", 1999.
2. PRESSMAN, Roger S. 1997. Ingeniería de Software. Cuarta edición.
3. SCHMULLER, Joseph. Aprendiendo UML en 24 horas
4. <[http://es.wikipedia.org/wiki/Portal\\_\(Internet\)](http://es.wikipedia.org/wiki/Portal_(Internet))> [Consulta: 25 de enero de 2010]
5. <<http://www.malagadnug.org/ficheros/MSFMartinLuisReq.pdf>>, [Consulta: 25 de Enero de 2010]
6. <<http://www.desarrolloweb.com/articulos/499.php>>, Publicado: 24/7/01, Por Miguel Ángel Álvarez, [Consulta: 25 de Enero de 2010]
7. <[http://es.wikipedia.org/wiki/Microsoft\\_SQL\\_Server](http://es.wikipedia.org/wiki/Microsoft_SQL_Server)>, Esta página fue modificada por última vez el 16:11, 24 Junio 2009. [Consulta: 25 de Enero de 2010]

8. <[http://es.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio\\_.NET](http://es.wikipedia.org/wiki/Microsoft_Visual_Studio_.NET)> [Consulta: 6 de Febrero de 2010]
9. <<http://es.wikipedia.org/wiki/ASP.NET>> [Consulta: 6 de Febrero de 2010]
10. <<http://www.alegsa.com.ar/Dic/.net%20framework.php>> [Consulta: 6 de Febrero de 2010]
11. <<http://www.trucosgratis.net/foro/index.php/topic,155.0.html>> [Consulta: 18 de Febrero de 2010]
12. <[http://www.elguille.info/NET/ADONET/SQL2005Express/creabase\\_usarscript\\_managementStudio.htm](http://www.elguille.info/NET/ADONET/SQL2005Express/creabase_usarscript_managementStudio.htm)> [Consulta: 18 Febrero de 2010]
13. <<http://www.infor.uva.es/~jvegas/cursos/buendia/pordocente/node23.html>> [Consulta: 18 de Febrero de 2010]

**ANEXOS**

# INDICE

1. INTRODUCCIÓN	1
1.1. ANTECEDENTES	14
1.2. PLANTEAMIENTO DEL PROBLEMA	15
1.3. SISTEMATIZACIÓN	17
1.3.1. Diagnóstico	17
1.3.2. Pronóstico	25
1.3.3. Control del Pronóstico	28
1.4. OBJETIVOS	30
1.4.1. Objetivo general	30
1.4.2. Objetivos específicos	30
1.5. JUSTIFICACIÓN	31
1.5.1. Justificación Teórica	31
1.5.2. Justificación Práctica	33
1.5.3. Justificación Metodológica	34
1.6. ALCANCE	35
1.7. ESTUDIOS DE FACTIBILIDAD	37
1.7.1. Factibilidad Técnica	37
1.7.2. Factibilidad Económica	45
1.7.3. Factibilidad Operacional	46
2. MARCO DE REFERENCIA	48
2.1. MARCO TEÓRICO	48

<b>2.2. MARCO CONCEPTUAL</b>	<b>53</b>
<b>2.3. MARCO ESPACIAL</b>	<b>71</b>
<b>2.4. MARCO LEGAL</b>	<b>72</b>
2.4.1 Disposiciones Legales	72
2.4.2. El software como objeto protegido	74
<b>3. METODOLOGIA</b>	<b>76</b>
<b>3.1. METODOLOGÍA DE INVESTIGACIÓN</b>	<b>76</b>
3.1.1. Unidad de Análisis	76
3.1.2. Tipos de Investigación	77
3.1.3. Métodos	77
3.1.4 Fuentes de información	78
3.1.5. Técnicas	78
3.1.6. Instrumento	78
<b>3.2. METODOLOGÍA INFORMÁTICA</b>	<b>82</b>
3.2.1. Metodología	82
3.2.2. Ciclo de Vida	82
3.2.3. Proceso de Desarrollo	84
<b>4. PROCESO DE DESARROLLO</b>	<b>86</b>
<b>4.1 PLAN DE DESARROLLO DE SOFTWARE</b>	<b>86</b>
4.1.1. Propósito	86
4.1.2 Alcance	87
4.1.3 Resumen	87
4.1.4 Vista general del proyecto	88
4.1.5 Organización del proyecto	96

4.1.6 Gestión del proceso	98
<b>4.2 Modelo de casos de uso del negocio</b>	<b>106</b>
4.2.1 Diagrama general de casos de uso del negocio	106
4.2.2 Descripción de los casos de uso del negocio	108
4.2.3 Diagramas de actividad	111
4.2.4 Diagrama de clases del negocio	114
4.2.5 Diagrama dinámico	118
4.2.6 Diagrama de componentes	119
4.2.7 Diagrama arquitectónico	120
4.2.8 Diagrama de colaboración	121
<b>4.3 Modelo de casos de uso</b>	<b>123</b>
<b>4.4 GLOSARIO</b>	<b>126</b>
<b>4.5. VISIÓN</b>	<b>128</b>
4.5.1 Introducción	128
4.5.2 Posicionamiento	130
4.5.3 Descripción de stakeholders (Participantes en el Proyecto) y usuarios	131
4.5.4 Descripción global del producto	135
<b>4.6 Especificación de casos de uso</b>	<b>138</b>
<b>4.7 ESPECIFICACIONES ADICIONALES</b>	<b>140</b>
<b>4.8 PROTOTIPOS DE INTERFAZ DE USUARIO</b>	<b>142</b>
<b>4.9 Modelo de Datos</b>	<b>154</b>
<b>4.10 Modelo de Implementación</b>	<b>155</b>
<b>4.11 Modelo de despliegue</b>	<b>156</b>

<b>4.12 CASOS DE PRUEBA</b>	<b>156</b>
4.12.1 Casos de prueba del caso de uso: Registrar ingresos	156
4.12.2 Casos de prueba del caso de uso: Registrar gastos	158
4.12.3 Casos de prueba del caso de uso: Generar Impuesto	160
4.12.4 Casos de prueba del caso de uso: Administrar Aplicación	161
<b>5. CONCLUSIONES Y RECOMENDACIONES</b>	<b>165</b>
<b>5.1. CONCLUSIONES</b>	<b>165</b>
<b>5.2. RECOMENDACIONES</b>	<b>168</b>
<b>BIBLIOGRAFÍA</b>	<b>169</b>
<b>ANEXOS</b>	<b>171</b>